



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

INLIGTINGSTEKNOLOGIE

RIGLYNE VIR PRAKTIESE ASSESSERINGSTAKE

2014

Hierdie riglyne bestaan uit 35 bladsye.

Wat is die PAT?

Die PAT is 'n programmatuur-ontwikkelingsprojek wat jou die geleentheid gee om jou programmeringsvaardighede asook jou begrip van die onderlinge verband tussen die verskillende areas van oplossingsontwikkeling, te toon.

Jy sal moet toon dat jy kennis en begrip van die programmatuur-ontwikkelingsiklus het deur analisering, ontwerp, konstruksie/implementering en toetsing van jou projek. Jy sal effektiewe gebruik van programontwerpgereedskap en -tegnieke wat jy bestudeer het, moet toon.

Na voltooiing van die projek sal jy die volgende as bewys moet inhandig:

- 'n verslag waarin jy (fase 1)
 - o 'n kort beskrywing van die doel en omvang van jou projek gee
 - o die navorsing/ondersoek wat jy in verband met die projek gedoen het, bespreek
 - o 'n analise van 'n moontlike oplossing verskaf
- 'n dokument wat die stelselontwerp uiteensit (fase 2)
- 'n werkende Delphi/Java-program, ten volle gedokumenteerd, wat die beplande oplossing implementeer (fase 3)

LET WEL: Daar sal van jou verwag word om jou program te demonstreer en te bespreek tydens 'n voorleggingsessie.

Puntetoekenning

Omdat die PAT 25% van jou finale punt vir IT tel, is dit van uiterste belang dat jy poog om werk van 'n hoë standaard te lewer.

Fase	Ontwikkelingsfase	Maksimum Punt
Fase 1	Analise	20
Fase 2	Ontwerp	50
Fase 3	Kodering en toetsing (50) Vlak van kompleksiteit (30)	80
Algemeen	Finale produk en indruk	20
Totaal:		180

LET WEL:

- Die PAT-punt is 'n verpligte komponent van die finale sertifiseringspunt vir alle kanidate wat vir Inligtingstegnologie geregistreer is.
- **Inhandigingsdatum:** Jou voltooide PAT moet nie later as drie weke voor die aanvang van die finale Graad 12-eksamen voorgelê en ingehandig word nie.
- Jou PAT sal deur vakkundiges gemodereer word en die kwaliteit van die werk sal deur Umalusi nagegaan word.

Die Probleem

ONDERWERP – Programmatuurstelsel vir 'n nie-winsgewende/liefdadigheidsorganisasie

As deel van 'n poging om by te dra tot die goeie werk wat nie-winsgewende en liefdadigheidsorganisasies soos CANSA (ondersteun kanker pasiënte), Die Rooikruis, kinderhuise, DBV (Dierebeskermingsvereniging), organisasie wat dowes, blindes ens. doen, moet jy 'n inligtingstelsel ontwikkel wat enigteen in jou omgewing wat by hierdie tipe organisasies betrokke is, sal help.

Kies 'n spesifieke nie-winsgewende organisasie en doen navorsing omtrent hulle inligtingstelselbehoeftes.

Die stelsel wat jy ontwikkel kan die organisasie van jou keuse ondersteun met aktiwiteite soos:

- Hou boek van hulle personeel, aktiwiteite en lede
- Lê inligting vas omtrent fondsinsamelingsgeleenthede
- Hou boek van hulle verskillende takke, donasies en ondersteuners
- Lê inligting vas omtrent bewusmakingsveldtogte
- Organiseer die bestel, verspreiding en inkomste wat gegenereer word uit bemerkingsmateriaal.

Jy word nie beperk deur die bogenoemde lys van funksionaliteite nie maar jy moet data en funksionaliteite (dienste) so kies dat jy 'n bruikbare en goed afgeronde toepassing kan skep.

Spesifieke vereistes:

- Die projek moet die belangrikste ontwikkelingsgereedskap insluit, m.a.w. 'n databasis en programmeringstaalkonstrukte wat goed geïntegreer is.
- Al die aspekte wat in die PAT-vereistes genoem is (sien bladsy 5), en fase 3 se nasienriglyne (Bylaag A).

Algemene programmeringsaspekte wat in jou projek geassesseer sal word:

- Programmeringstyl
- Grafiese gebruikerskoppelvlak (GUI)
- Gebruik van goeie mens-rekenaar interaksie- en programmatuurontwikkelingsbeginsels
- Toevoer- en afvoerbestuur
- Funksionaliteit van die program
- Vlak van deskundige programmering
- Hoe robuust die program is, insluitend die gebruik van defensiewe programmering
- Of die projek die oorspronklike doelstellings en mikpunte bereik
- Interne dokumentasie om dele van die program te verduidelik

LET WEL: Jou finale program moet 'n **enkele**, logies verwante program wees. Vir projekte wat uit twee of meer onverwante programme bestaan, sal slegs punte vir een van die programme toegeken word omdat slegs een van die programme as die eintlike projek beskou sal word.

Oorsig

FASE 1 – ANALISE

Tydens hierdie fase moet jy bewys dat jy 'n goeie en deeglike gebruikersbehoefte-analise gedoen het om te bepaal wat die gebruikers van die stelsel wil hê dat die stelsel moet kan doen. Die volgende kan as 'n riglyn gebruik word:

- Bepaal die vereistes van die kliënt/gebruiker – doen navorsing indien nodig en/of hou onderhoude.
- Doen navorsing oor moontlike oplossings deur na bestaande stelsels te kyk.
- Doen verslag oor jou navorsing/ondersoek deur duidelike verwysings/bewyse te verskaf.
- Stel gedetailleerde vereistes op en spesifiseer wat jou oplossing sal bied om aan die vereistes te voldoen.

LET WEL: Die **gebruiker** is die teikengehoor en bepaal dus die behoeftes of vereistes.

FASE 2 – ONTWERP

Die doel van fase 2 is om te bepaal **hoe** die program/stelsel aan die vereistes sal voldoen. 'n Goed beplande oplossing met die gedetailleerde spesifikasies moet voorsien word.

Die volgende moet gedoen word:

- Maak vereistes duidelik deur spesifiek aan te dui hoe jou program/oplossing elke vereiste sal nakom.
- Gebruik die nodige gereedskap en ontwerp die databasis, GUI en program deur die presiese spesifikasies van elkeen te gee.
- Dui duidelik die logiese vloei van die program aan en die navigasie tussen die skerms.

LET WEL:

Die volgende gereedskap behoort/kan in die ontwerp-proses gebruik word:

- o TVA-diagramme, TOG-tabelle, EV-diagramme.
- o Vloeddiagramme, UML-diagramme (klasdiagramme, aktiwiteitsdiagramme en gebruik-geval diagramme).
- o Stelsel- en databasis-ontwerpgereedskap wat op die Internet beskikbaar is.

FASE 3 – KODERING EN TOETSING

Die doel van fase 3 is om die ontwerp te implementeer deur die kode te skryf en die program te toets:

- Skryf die programmeringskode om die ontwerp te implementeer en die program te voltooi
- Toets en ontfout die program
- Maak stukke kode duidelik deur kommentaar by te voeg
- Skryf projeknotas vir die program
- Demonstreer jou program en beantwoord vrae omtrent die program en die kode tydens 'n ondervragingsessie

PAT-Vereistes

Die projek moet die volgende insluit:

- Databasismanipulering deur programmeringstaalkonstrukte,
- 'n Multi-vorm/multi-skerm GUI met goeie funksionaliteit en bruikbaarheid, gebaseer op goeie HCL-beginsels
- Die gebruik van 'n tekslêer vir toevoer-afvoer doeleindes, bv. om datastrukture te vul en verslae te verskaf.

Databasis

Die databasis-komponent moet:

- meer as een gekoppelde tabel (verwante tabelle) gebruik
- voldoende volumes data en 'n verskeidenheid tipes velde betrek
- eie programmeringskode gebruik (in kombinasie met SQL) om toegang daartoe te kry en dit te manipuleer.

LET WEL: Die munt wat jy vir jou projek kry sal grootliks beïnvloed word deur die kwaliteit van die programmeringskode wat die data suksesvol manipuleer op die beste moontlike manier om aan die vereistes van die gebruiker te voldoen. Hoeveelheid sal nie verskeidenheid, effektiwiteit en kwaliteit vervang nie.

Tekslêer

Jou toepassing moet tekslêers gebruik om toevoer te aanvaar vanuit 'n lêer om data wat op 'n ander plek ingesleutel kon gewees het, te bestuur. Dit behoort meganismes in te sluit om data oor te dra en te verwerk tussen jou databasis en die tekslêer/s.

Die data uit die tekslêer kan gebruik word om:

- berekeninge en manipulering in kombinasie met data uit die databasis te doen
- bestaande data op te dateer
- rekords in 'n databasis in te voeg.

Ten minste een verslag moet in 'n tekslêer-formaat voorsien word.

GGK ('GUI')

Die grafiese gebruikerskoppelvlak (GGK) moet ten minste:

- drie vorms/skermse hê en
- twee komponente (verskillende tipes) dinamies skep.

Datastrukture (programmering)

Daar word van jou verwag om een of meer klasse te ontwerp wat attribute, konstruktors en metodes met parameters en terugstuur-tipes bevat.

Inkorporeer ook jou eie metodes, bv. om data te valideer of om data wat in die databasis gestoor is te manipuleer.

Verdere vereistes

Pas goeie programmeringspraktyke toe, soos die volgende:

- Gebruik beskrywende name vir veranderlikes, datastrukture, velde, komponente, ens.
- Stel goed gestruktureerde en leesbare programmeringskode saam.
- Pas modulêre programmering toe (ontwerp metodes om take uit te voer en groepeer metodes in klasse).
- Skryf kommentaar/beskrywende notas om stukke kode te verduidelik. Gee meer gedetailleerde kommentaar oor die manier waarop toevoerveranderlikes/datastrukture en afvoerveranderlikes/komponente gebruik en geïnterpreteer word.

Skryf projeknotas

- Hierdie notas moet beskryf hoe om interaktief met die program te werk. Die notas moet ook enige bekende foute of probleme beskryf.
- Projeknotas kan geskryf word as deel van die hulpfunksie van die program.

Wat jy nodig het om die PAT te kan doen

Om die PAT te kan doen, het jy die volgende nodig:

- Java/Delphi programmeringsprogrammatuur insluitend 'n GGK (GUI) *IDE* (Geïntegreerde ontwikkelingsomgewing)
- 'n 'Office'-pakket met die volgende programmatuur:
 - o Woordverwerker-programmatuur
 - o Databasis-programmatuur
- Toegang tot die Internet om data en inligting te kry
- Toegang tot ander bronne soos gedrukte media (bv. tydskrifte, koerante, brosjures, handboeke) of ander elektroniese materiaal (bv. e-boeke, e-artikels).
- Toegang tot fasiliteite om harde kopieë om te skakel na elektroniese dokumente, bv. skandeerders, digitale kamera
- Stoomedia om jou werk elektronies te stoor en te rugsteun, bv. geheuestokkie, herskryfbare CD/DVD.

Wanpraktyk

Omdat die PAT 'n individuele projek is wat deel van jou finale promosiepunt is mag jy nie:

- Hulp kry van ander sonder om erkennig te gee nie
- Ander toelaat om programmeringskode vir jou te doen nie
- Werk inhandig wat nie jou eie is nie
- Jou werk met ander leerders deel nie
- Werk insluit wat direk van boeke, die Internet of ander bronne af gekopiëer sonder om erkenning te gee nie.

Die bostaande aksies is wanpraktyke waarvoor daar 'n straf toegepas sal word afhangende van die erns van die oortreding.

LET WEL: As jy werk uit ander bronne gebruik mag dit nie 30% van die werk wat jy inhandig oorskry nie

Geen inhandiging nie

Jy sal tot drie weke voor die aanvang van die finale eksamen die einde van die jaar gegee word om uitstaande werk in te handig of jou PAT projek voor te lê.

As jy versuim om aan die Praktiese Assesseringsprojek se vereistes te voldoen, sal 'n nul ('0') aan jou toegeken word vir die PAT-komponent van IT.

Instruksies vir Fase 1

Daar word van jou verwag om die probleem/taak wat gedoen moet word goed te verstaan. Doen goed navorsing omtrent die onderwerp sodat jy 'n goeie idee het van wat verwag word van die programmatuur wat jy moet ontwikkel

Kies 'n maatskappy of toepassing om die spesifikasies wat in hierdie dokument beskryf word, op toe te pas. Hou onderhoude of analiseer die maatskappy se behoeftes of toepassingsvereistes noukeurig om te bepaal wat die program-oplossing moet doen en voorsien.

Die uitset van hierdie fase is 'n duidelik verduideliking van wat die probleem/taak is en wat die oplossing in staat moet wees om te doen in terme van die behoeftes van die gebruiker (maatskappy)/taak en hulle gestelde doelstellings.

DEFINIEER DIE TAAK

Gee 'n kort beskrywing (± 150 woorde) in jou eie woorde van die doel van die probleem/taak en hoe die projek die probleem gaan oplos in algemene terme. Met ander woorde die beskrywing behoort geskryf te word om die bestuurder van die maatskappy te oortuig om jou te huur om die programmatuur wat vereis word, te ontwikkel. Die bestuurder moet besef dat

- jy die probleem verstaan
- dat jou oplossing die probleem sal oplos

Dit moet ook beskryf hoe jou program aan die PAT-vereistes/spesifikasies sal voldoen.

Die beskrywing moet 'n oorsigtelike beeld gee van wat die doel en omvang van die projek is maar nie die detail nie.

DOEN NAVORSING

Die navorsingstadium is waar jy feite omtrent die aard van jou program wat jy gaan ontwikkel, insamel.

Jou navorsing moet jou help om die tipe program wat jy gaan ontwikkel uit te klaar, bruikbare voorbeelde wat jou kan lei voorsien en duidelike begrip hoekom die spesifieke tipe program geskik is vir hierdie projek voorsien.

Die uitkoms van die navorsing is 'n opsomming (± 1200 woorde). Die opsomming moet in duidelike, ondubbelsinnige taal geskryf word en kan diagramme wat meer verduidelik insluit.

DOEN DIE ANALISE

Die doel is om te bepaal wie die gebruiker is, wat die data en verwerkingsvereistes van die stelsel, met inagneming van die menslike aspekte (bv. geskikte gebruikerskoppelvlak) en fisiese omgewing (bv. apparatuur-vereistes). Dit moet spesifiseer **WAT** nodig is (nie HOE nie).

Spesifiseer en dokumenteer die data- en verwerkingsvereistes vir die oplossing van die probleem:

- Identifiseer die moontlike gebruiker(s)
- Identifiseer gebruiker se behoeftes en aanvaarbare beperkinge
- Databron(ne) en -bestemming(s)
- Regverdiging van die voorgestelde oplossing

HANDIG IN

Handig 'n dokument in wat die volgende bevat:

- die projekbeskrywing (±150 woorde)
- 'n Verslag wat die volgende bevat
 - o 'n opsomming van die navorsing/ondersoek (±1200 woorde)
 - o die analise van gebruikersbehoefte
 - o die aanvaarbaarheidstoetse

Instruksies vir Fase 2

Dit is hier waar jy die detail beplan en inligting voorsien oor **HOE** jy te werk sal gaan om die probleem op te los.

Die doel is om die algehele ontwerp wat aan die vereistes voldoen, te spesifiseer en te dokumenteer deur programmatuurontwerpgereedskap soos TVA-diagramme, TOG-tabelle ('TOE charts'), vloei-diagramme, EV-diagramme ('ER diagrams'), UML-diagramme (gebruik-geval en klasdiagramme), duidelike notas te gebruik.

Die uitkoms is 'n plan wat 'n hoëvlak oorsig bied van **hoe** die oplossing saamgestel sal word deur van met pseudokode/diagramme (of toepaslike alternatief) gebruik te maak wat verduidelikende notas bevat. Die plan moet die belangrikste blokke/aspekte in die voorgestelde oplossing toon.

Spesifiseer en dokumenteer

- Die probleemoplossingsmetode
- Die funksies van die belangrike dele van die program/stelsel
- Die onderlinge verband tussen die verskillende dele van die program/stelsel
- Die algoritmes, datatipes en datastrukture asook enige ander vereistes van die oplossing
- Die effektiwiteit van die voorgestelde oplossing sover dit voldoening aan die vereistes van die probleem aangaan

ONTWERP DIE DATABASIS

Die doel is om 'n databasis te ontwerp wat as databron kan dien asook om data in die databasis met programmeringsinstruksies te manipuleer.

Die databasis moet data aan die program voorsien om verwerk te word en verslae te voorsien.

Die Delphi/Java-program moet die inhoud van databasistabelle kan manipuleer, bv. bywerk/redigeer/skrap/byvoeg van data, resultate van navrae voorsien, verslae voorsien, ens.

ONTWERP DIE GRAFIESE GEBRUIKERSKOPPELVLAKE ('GUI')

Die doel is om 'n GGK ('GUI') te ontwerp wat goeie rekenaar-mens-koppelvlak (*HCI*)-beginsels in ag neem, wat foute as gevolg van ongeldige invoer voorkom en wat die hoeveelheid inligting wat 'n gebruiker moet intik tot die minimum beperk.

Gebruik *HCI*-ontwerpbeginsels en ontwerp 'n GGK wat die volgende in ag neem:

- Die gebruiker – tipe gebruiker en konteks
- Gebruiker se behoeftes, bruikbaarheid ('*usability*')
- Dialoog – moet toepaslik, eenvoudig en duidelik wees
- Gebruik en aanbieding van ikone – goed gekies en relevant, goed geplaas en die doel daarvan duidelik
- Kleur – gebruik en kombinasie van kleure word toegepas
- Terugvoer – netjies duidelik en goed aangebied

- Nuttige foutboodskappe
- Uitgange – duidelik aangedui, goed geplaas
- Kortpadsleutels
- Vloei van inligting op die skerm – bo na onder en links na regs
- Sinvolle gebruik van spasie op die skerm.

Voorsien voorbeelde van beplande datavaslegging- en datatoevoerontwerpe (skermkopieë van 'n prototipe van die projek mag gebruik word maar moet beskrywende notas bevat) en van beplande afvoer-ontwerpe.

ONTWERP DIE OPLOSSING (VLOEI, ALGORITMES, DATATYPES/STRUKTURE, ENS.)

Gebruik geskikte gereedskap om die oplossing in geheel te ontwerp met inagneming van al die onderlinge dele en die onderlinge verwantskappe tussen die verskillende dele van die stelsel.

- Beskryf die modulêre struktuur van die program/stelsel
- Definieer die datavereistes
 - o Strukture
 - o Lêerorganisasie en -verwerking (bv. tekslêers)
 - o Validering wat benodig word
- Identifikasie van prosesse en toepaslike algoritmes vir datatransformasie
- Diagramme/definisies en detail van klasse, objekte, hul attribute en metodes
- Beskryf beplande maatreëls om die sekuriteit en integriteit van data te verseker

ONTWIKKEL 'N TOETSSTRATEGIE

Kies en dokumenteer geskikte toetsdata met verwagte resultate vir normale (tipiese) data, foutiewe data en grensdata (uiterstes).

HANDIG IN

Handig 'n dokument wat die volgende voorsien:

- 'n Beplande databasisontwerp
- G GK-ontwerp
- Ontwerp van 'n klas of klasse
- Algehele oplossingontwerp
- Toetsstrategie

Instruksies vir Fase 3

Dit is hier waar jy jou ontwerp implementeer deur van toepaslike programmeringsgereedskap (programmeertaal, databasisprogrammatuur, GOO ('IDE'), ens.) en -tegnieke gebruik te maak om 'n oplossing vir die probleem te skep.

Nadat jy jou projek voltooi het, moet jy die program demonstreer en vrae oor jou program, die proses en die kode beantwoord.

ONTWIKKEL DIE DATABASIS

Implementeer die ontwerp en skep die databasis deur die toepas van toepaslike tegnieke.

Maak seker dat die databasis korrek met die program koppel en op 'n betekenisvolle en effektiewe wyse, wat die oplossing ondersteun, met die program saamwerk.

ONTWIKKEL DIE G GK ('GUI')

Implementeer die ontwerp deur die G GK te ontwikkel en gebruik toepaslike komponente wat gemaklike gebruik en navigasie sal verseker. Die gebruiker moet 'n aangename ervaring hê wanneer hy/sy die program gebruik.

SKRYF DIE KODE

Gebruik die beplanningsdokumente van fase 1 en fase 2 en skryf die kode vir al die eenhede/dele

Gebruik goeie programmeringstegnieke en -strukture.

Implementeer effektiewe algoritmes en goeie defensiewe programmeringstegnieke om 'n robuuste program te verseker.

Dokumenteer die kode sodat ander mense in staat sal wees om die program te interpreteer en sal verstaan wat individuele stukke kode doen.

DOKUMENTEER DIE PROGRAM

Gebruik enige toepaslike fasiliteit van die programmeringstaal en skryf projeknotas wat vir die gebruiker toeganklik is, om te verduidelik hoe om die program te gebruik.

Die notas moet ook enige bekende programfoute of probleme beskryf.

Voeg kommentaar by om stukke kode te verduidelik

TOETS DIE PROGRAM/STELSEL

Toets die program/stelsel deur duidelik gedefinieerde, tipiese data, foutiewe data en grensdata (ekstreme/uiteerstes) te gebruik.

HANDIG IN

Handig in:

- Die voltooide Delphi/Java-projek, insluitend die kommentaar en projeknotas.
- Die egtheidsverklaring (Sien **Bylaag E**)

ONDERVRAGING

Demonstreer die program vir evaluering en ondervraging.

Riglyne vir die demonstrasie van die program:

- Die onderwyser sal datums en tye vir die demonstrasies skeduleer. Ongeveer 15 minute per projek sal toegelaat word.
- Jy moet voor die demonstrasie alle dokumentasie inhandig – ten minste een week vooraf.
- Die demonstrasie moet elektronies op 'n rekenaar gedoen word.
- Vir evalueringsdoeleindes moet jy jou program uitvoer en al die kenmerke van die program aan jou onderwyser uitwys.
- Die onderwyser kan van jou vereis om toetsprosedures uit te voer om seker te maak dat die hele program reg werk.
- Die onderwyser kan tydens die demonstrasie die nasienblad vir Fase 3 as 'n riglyn gebruik en punte dienooreenkomstig toeken.
- As deel van die demonstrasie, sal die onderwyser op 'n ewekansige manier dele van die programkode identifiseer en jou vra om die doel en werking daarvan te verduidelik. Dit word gedoen om seker te maak dat jy die kode self geskryf het. 'n Soortgelyke prosedure sal tydens moderering gevolg word. Indien jy nie die kode in die projek kan verduidelik nie, kan geen punte vir die projek toegeken word nie.
- Jy moet die elektroniese kopie van die projek wat jy gedemonstreer het inhandig. Die onderwyser sal hierdie kopie gebruik om enige uitstaande punte toe te ken om sodoende die punt te finaliseer.

STERKTE!

Bylaag A: Assesseringsinstrument

Fase 1: Naam van leerder:							
Scenario	4	3	2	1	0		
Scenario (Kort beskrywing ±150 woorde)	Taak is duidelik gestel (doel en gehoor) Deeglike begrip van wat die probleem/taak behels Alle aspekte hanteer.	Taak is duidelik gestel (doel en gehoor) Deeglike begrip van wat die probleem/taak behels Omtrent alle aspekte hanteer.	Doel en gehoor nie so duidelik nie Tekortkominge in begrip en nie alle aspekte hanteer nie.	Vaag, onseker oor doel van die program. Minimale begrip en min van die aspekte hanteer.	Totaal onvoldoende of nie van toepassing Swak of geen hantering van aspekte nie	4	
Onderzoek/ navorsing	4	3	2	1	0		
Verslag oor sleutelareas wat die program sal dek		Uitgebreide navorsing is gedoen. Verduidelik alle sleutelareas. Opsomming wat 'n goeie aanduiding gee van wat die program behoort te doen Toon deeglike begrip	Aanvaarbare hoeveelheid navorsing is gedoen Bespreek meeste van die sleutelareas Aanvaarbare opsomming Toon redelike begrip	Beperkte hoeveelheid navorsing is gedoen Vaag, te min dekking van sleutelareas. Kort onvolledige opsomming Toon minimale begrip	Geen bewys van navorsing nie Geen sleutelares of verkeerd en irrelevant of nie gedoen nie	3	
Afsluiting		Uitstekend, duidelike rigtinggewend vir die projek, bv. definieer duidelik die reikwydte van die program Duidelike oorsig van 'n baie toepaslike moontlike oplossing. Deeglike insig en begrip	Voldoende rigting – nie altyd afgelei van ondersoek, reikwydte en doel nie altyd duidelik in alle aspekte nie. Voldoende insig	Vaag, rigting onduidelik min verband met ondersoek of reikwydte en moontlike oplossing nie geskik nie Min insig	Nie afgelei van die ondersoek of relevant tot die onderwerp nie Nie rigting vir die projek nie of geen afsluiting nie	3	
Struktuur van die verslag		Goed-gestruktureerde verslag. Voorsien bv. relevante skermkote, drukstukke, ens Sluit alle aspekte in wat uitgelig is in die ondersoek-afdeling	Aanvaarbare struktuur 'n Paar skermkote, drukstukke ens. Nie altyd relevant nie Sluit feitlik alle aspekte in wat in die ondersoek- afdeling uitgelig is.	Swak struktuur Geen skermkote, drukstukke ens. Nie altyd relevant nie Sluit slegs 'n paar aspekte in wat in die ondersoek-afdeling uitgelig is	Geen verslag of nie relevant tot die aspekte wat in die ondersoek-afdeling uitgelig is nie	3	
Verwysings			Alle verwysings (ten minste 3) maak gebruik van die Harvard/APA styl	Sommige (slegs 2) verwysings is ingesluit of verkeerde styl	Geen verwysings is ingesluit nie.	2	

Gebruikersvereistes	4	3	2	1	0		
Rol, aktiwiteit, vereistes en beperkings	Rol, aktiwiteit, vereistes en beperkings van ten minste 3 verskillende tipe gebruikers van die stelsel is bespreek. Goed gedokumenteer maar kan effens daarop verbeter	Sommige tekortkominge in die bespreking van die rol, aktiwiteit, vereistes en beperkings van ten minste 3 verskillende tipe gebruikers van die stelsel is bespreek. Goed gedokumenteer maar kan effens daarop verbeter	Tekortkominge in die bespreking van die rol, aktiwiteit, vereistes en beperkings gebruikers bv. afdelings is uitgelaat. Slegs 2 verskillende tipes gebruikers van die stelsel is bespreek. Nie goed gedokumenteer nie, maar aanvaarbaar.	Ernstig tekortkominge in die bespreking van die rol, aktiwiteit, vereistes en beperkings gebruikers bv. afdelings is uitgelaat. Slegs 1 gebruiker van die stelsel is bespreek. Swak gedokumenteer – nie aanvaarbaar nie	Nie gedoen nie of verkeerd of irrelevant	4	
Aanvaardingstoetse	Toetse vir alle vereistes is duidelik gedefinieer	Toetse vir feitlik alle vereistes is gedefinieer	Toetse vir meeste van die vereistes is gedefinieer. Nie altyd goed gedefinieer nie	Toetse vir 'n beperkte getal van die vereistes is gedefinieer. Swak gedefinieer	Geen toetse gedefinieer nie of verkeerd	4	
Gebruikersgeval-diagramme		Beskryf duidelik wat die stelsel doen om die behoeftes van alle gebruikers te akkommodeer Toon duidelike vereistes en toetsgevalle in alle gevalle	Beskryf duidelik wat die stelsel doen vir alle gebruikers maar klein tekortkominge Toon vereistes en toetsgevalle maar geringe tekortkominge	Beskryf wat die stelsel doen in sommige gevalle Toon vereistes en toetsgevalle in sommige gevalle	Geen gebruikersgeval-diagramme nie of verkeerd	3	
Voorlegging	4	3	2	1	0		
Tydsbestuur – Fase 1	Hou by afsnydatum, alle werk gedoen	Een dag laat maar volledig	Een dag laat en amper volledig	Een dag laat en afdelings nie gedoen nie of onvolledig	Meer as een dag laat of nie gedoen	4	
Totaal						30	

Kommentaar/terugvoer:

Naam van onderwyser: _____ Onderwyser se handtekening: _____ Datum: _____

Fase 2: Naam van leerder:						
Databasis ontwerp	3	2	1	0		
Keuse van velde	Uitstekende keuse van velde Alle velde dra by tot die oplossing. Geen onnodige velde nie	Meeste velde dra by tot die oplossing OF EEN onnodige veld	Meer as een veld wat nie bydra tot oplossing nie OF meer as een onnodige veld	Geen databasis of verkeerd of nie relevant nie	3	
Veldtipes en -grootte	Alle velde goed gekies wat tipe en grootte aan betref	Meeste velde goed gekies wat tipe en grootte aan betref	Swak seleksie van velde, veldtipes en groottes in meeste gevalle	Geen databasis nie	3	
Geskiktheid – Tabele & verwantskappe	Geskikte aantal tabelle vir die oplossing Geskikte verwantskappe tussen die tabelle. Sleutels reg geïdentifiseer in al die tabelle. Goeie versameling van voorbeeld-data	Ten minste twee tabelle Geen swak of verkeerde gebruik van verwantskappe nie Sleutels verkeerd gebruik in sommige gevalle. Beperkte versameling van voorbeeld-data	Slegs een tabel/slegs die velde genoem – geen verwantskappe nie Regte primêre sleutel Swak versameling van voorbeeld-data	Slegs een tabel/slegs die velde genoem – geen verwantskappe nie Geen/verkeerde primêre sleutel Of geen databasis	3	
Rol van DB Beskryf hoe DB gemanipuleer sal word bv. binne 'n datastel, toegang tot velde en rekords, navigasie in rekords, ens. Rol van manipulasie in program beskryf/gemotiveer	Beskryf duidelik hoe DB gemanipuleer sal word in die program en hoe dit sal bydra tot die oplossing Alle manipulasie duidelik gemotiveer	Manipulasie en interaksie meestal goed beskryf en redelik gemotiveer. Meestal geskik om vereistes na te kom	Manipulasie nie goed gemotiveer nie of meestal nie geskik om vereistes na te kom nie.	Geen databasis of verkeerd of nie relevant nie	3	
'GUI' ontwerp	3	2	1	0		
'HCI' beginsels Neem 'GUI'-ontwerp in ag: • Doel van program & gebruiker • Standaard GUI-ontwerp beginsels • Maklik om te gebruik en logiese vloei • Navigasie duidelik aangedui • Vriendelike dialoog • Hulp	Goed ontwerpte 'GUI' met inagneming van al 6 aspekte	Goed ontwerpte 'GUI' met inagneming van die meeste van ten minste 4 aspekte	Beperkte 'GUI' ontwerp met inagneming van ten minste 3 aspekte	Swak 'GUI' ontwerp met inagneming van minder as 3 van die aspekte	3	

Komponente	Mees toepaslike komponente in alle gevalle. Uitstekende uitleg Alle keuses duidelik gestaaf	Toepaslike komponente gebruik in meeste gevalle Bevredigende uitleg Sommige keuses gestaaf	Toepaslike komponente is gebruik is 'n paar gevalle, Uitleg nie bevredigend nie Meeste keuses nie gestaaf nie	Ontoepaslike komponente gebruik in meeste gevalle of keuses nie gestaaf nie.	3	
Datastrukture gebruik (Databasis uitgesluit)	3	2	1	0		
Keuse van Datastrukture, bv. skikkings, teksleërs, veranderlikes, ens. (Hoe data gestoor word)	Uitstekende verskeidenheid van die mees toepaslike datastrukture. Alle keuse dra duidelik by tot die oplossing en is duidelik gestaaf.	Verskeidenheid van die toepaslike datastrukture. Een of twee datastrukture kon met iets meer toepaslik vervang word. Meeste keuses gestaaf	Verskeidenheid van die toepaslike datastrukture. Meer as twee datastrukture kon met iets meer toepaslik vervang word. Enkele keuses gestaaf	Nie beskryf of gestaaf nie of geen verskeidenheid	3	
OOP	3	2	1	0		
Klasse en Objekte	Groot getal klasse. Goed-gedefinieer insluitend attribute en metodes. Potensiële volume data is beduidend	'n Aanvaarbare getal klasse. Redelik goed gedefinieer, geringe tekortkominge in definisie van entiteite.	Beperkte getal klasse gedefinieer. Ernstige tekortkominge in definisie van klasse.	Geen klasse gedefinieer nie of totaal verkeerd of irrelevant	3	
Klasdiagramme	Klasse is duidelik beskryf, hulle attribute, bewerkings (metodes) en verwantskappe tussen die klasse in alle gevalle	Klasse is duidelik beskryf, hulle attribute, bewerkings (metodes) en verwantskappe tussen die klasse in meeste gevalle maar geringe tekortkominge	Klasse is beskryf, hulle attribute, bewerkings (metodes) en verwantskappe tussen die klasse in 'n paar gevalle – groot tekortkominge	Geen klasse nie of totaal verkeerd	3	
TVA ('IPO') ontwerp	3	2	1	0		
Toevoer (Hoe toevoer verkry en bestuur sal word)	Alle toevoer duidelik beskryf in terme van hoe die toevoer verkry sal word, hoe bronne van toevoer gebruik sal word en die formaat van die toevoer bv. tipe en grootte	Meeste toevoer is beskryf. Geringe tekortkominge in beskrywings	Sommige toevoer duidelik beskryf Beperkte beskrywings	Geen toevoervereistes beskryf nie of verkeerd	3	
Verwerking (Hoe verwerking bestuur sal word)	Verwerking/manipulasie duidelik beskryf in terme van hoe data verwerk/manipuleer moet word (algoritmes, formules, ens.)	Meeste verwerking/manipulasie duidelik beskryf Geringe tekortkominge	Slegs sommige verwerking/manipulasie beskryf. Meestal beperk, onvolledig of verkeerd beskryf	Verwerking/manipulasie nie beskryf nie Verkeerd of irrelevant	3	

Afvoer (Hoe afvoer bestuur sal word)	Afvoer duidelik beskryf in terme van hoe dit vertoon sal word, hoe bronne van afvoer gebruik sal word, asook die formaat, tipe en grootte van die afvoer	Meeste afvoer duidelik beskryf. Geringe tekortkominge	Slegs sommige afvoer is beskryf. Meestal beperk, onvolledig of verkeerd	Afvoer vereistes nie beskryf nie. Verkeerd of irrelevant	3	
Fouthantering	3	2	1	0		
Validasie (Hoe integriteit van toevoer, verwerking en afvoer hanteer sal word)	Beskryf duidelik toepaslike, betekenisvolle, effektiewe • tegnieke om TVA integriteit te verseker • validasie/fout-vang /hantering vir alle TVA • foutboodskappe geassosieer met validasie/foutvang	Tegnieke vir validasie/fout-vang meestal toepaslik en betekenisvol vir TVA	Tegnieke vir validasie/fout-vang slegs in sekere gevalle toepaslik en betekenisvol vir TVA. Slegs een of twee keer beskryf.	Tegnieke vir validasie/fout-vang glad nie beskryf nie of glad nie toepaslik nie	3	
Toetsing		2	1	0		
Toets data en verwagte resultate		Volledige stel van die mees toepaslike toetsdata om te gebruik, wat die verwagte resultate insluit	Minimale toetsdata of meestal nie toepaslik nie of minimale verwagte resultate ingesluit	Geen toetsdata, of toetsdata totaal onvanpas, of geen verwagte resultate	2	
Algehele beplanning	3	2	1	0		
Beplanning	Verskaf 'n deeglike hoë-vlak oorsigplan, wat duidelik wys hoe alle aspekte van die probleem opgelos sal word. Plan sluit goeie beskrywende notas saam met pseudokode/diagramme/kaarte (of geskikte alternatiewe) wat al die hoofblokke binne die voorgestelde oplossing toon	Verskaf 'n aanvaarbare hoë-vlak oorsigplan, wat 'n redelike poging bevat om te wys hoe sommige aspekte van die probleem opgelos sal word. Plan met goeie notas by pseudokode/diagramme/kaarte (of geskikte alternatiewe) wat die meeste hoofblokke binne die voorgestelde oplossing toon	Verskaf 'n beperkte hoë-vlak oorsigplan, minimale poging, baie tekortkominge. Plan sluit pseudokode/diagramme/kaarte, maar dui net sommige van die hoof blokke binne die voorgestelde oplossing in	Geen plan nie, of plan baie vaag en verwarrend. Geen beskrywende notas of enige hoofblokke aangedui nie.	3	

Gebruik van sagteware ingenieurshulpmiddels	3	2	1	0		
Programmatuurbepalingsinstrumente	Alle hulpmiddels, 'TOE chart', ERD en TVA kaarte toepaslik gebruik	Meeste van die hulpmiddels (ten minste 2) toepaslik gebruik	Sommige hulpmiddels toepaslik gebruik (ten minste 1)	Geen hulpmiddels gebruik of hulpmiddels nie toepaslik gebruik nie	3	
Aanbieding	3	2	1	0		
Tydsbestuur – Fase 2	Hou by afsnydatum – alle werk gedoen	Hou nie by afsnydatum nie – 1 dag laat – alle werk gedoen	Hou nie by afsnydatum nie – 2 dae laat of alle werk nie gedoen nie	Hou nie by afsnydatum nie – meer as twee dae laat of meeste van die werk nie gedoen nie	3	
Totaal					50	

Kommentaar/terugvoer:

Naam van onderwyser: _____ Onderwyser se handtekening: _____ Datum: _____

Implementering

Fase 3: Naam van leerder:							
Program aspekte	4	3	2	1	0		
Algoritmes Wat doen dit? Hoe goed word dit gedoen?	Alle oplossing algoritmes wat gebruik word in oplos van die probleem is die mees toepaslike en effektiewe, bv. geneste IF-stelling gebruik i.p.v. veelvuldige IF-stellings en werk korrek. Verbeter die projek	Toepaslike oplossing algoritmes gebruik en effektief, met enkele klein tekortkominge.	Meeste oplossing algoritmes is toepaslike en effektief	Meestal ontoereikende oplossing algoritmes en is oneffektief	Totaal ontoereikende oplossing algoritmes. Oplossing is oneffektief	4	
Beheerstrukture (voorwaardes, herhaling, ens.)	Gebruik toepaslike en effektiefste beheerstrukture om probleem in alle gevalle op te los	Gebruik toepaslike en effektiefste beheerstrukture om probleem in alle gevalle op te los, geringe tekortkominge	Toepaslike en effektiewe gebruik van beheerstrukture in meeste gevalle	Ontoepaslike en oneffektiewe gebruik van beheerstrukture in sommige gevalle	Totaal ontoepaslik en oneffektief	4	
Datastrukture (gebruikersgedefinieerd, DB uitgesluit)	Mees toepaslike en effektiefste datastrukture, korrek gebruik (bv. skikkings, tekslêers, ens.) om probleem in alle gevalle op te los	Mees toepaslike en effektiefste datastrukture, korrek gebruik om probleem in alle gevalle op te los, met geringe tekortkominge	Toepaslike en effektiewe datastrukture in meeste gevalle	Toepaslike en effektiewe datastrukture in sommige gevalle	Totaal ontoepaslik en oneffektief, of nie gebruik nie	4	
Interaktiwiteit/Datavloei (gebruikersgedefinieerde parameteroordrag)	Uitstekende/vaardige interaksie tussen modules, met parameter oordrag in almal	Meestal goeie, vaardige interaksie tussen modules, met parameter oordrag	Sommige interaksie tussen modules, met parameter oordrag	Beperkte interaksie tussen modules/dele ('units')	Geen interaktiwiteit nie	4	
Toevoer	Uitstekende verskeidenheid, mees toepaslike, effektiewe toevoer strategieë (databasis, tekslêers, gebruikerstoevoer) gebruik in alle gevalle.	Goeie verskeidenheid, toepaslik en effektief gebruik in alle gevalle, met geringe tekortkominge	Toepaslik en effektief gebruik in meeste gevalle. Beperkte verskeidenheid (bv. slegs twee tipes)	Toepaslik en effektief gebruik in sommige gevalle. Beperkte verskeidenheid (geen verskeidenheid nie, slegs een tipe)	Totaal ontoepaslik of oneffektief	4	

Afvoer (gekodeer)	In alle gevalle: Mees toepaslike vertoon, goed geformateer/leesbaar/gestruktureerd/verstaanbaar, bv. opskrifte en weer op bl./skerm wat volg, waar nodig. Geen logikafoute nie. Alle verwerkingsresultate korrek.	In alle gevalle: Mees toepaslike vertoon, goed geformateer/leesbaar/gestruktureerd/verstaanbaar maar met geringe tekortkominge in enkele geval. Een klein logiese fout. Een resultaat is problematies	In meeste gevalle: Toepaslike vertoon Sommige logiese foute Sommige resultate is nie korrek nie.	In sommige gevalle: Toepaslike vertoon Baie logiese foute Moeilik om afvoer te lees. Baie foute in resultate	Baie logiese foute Meeste resultate is foutief/slegs enkele van die vereiste resultate word gelewer	4	
Defensiewe programmering; Data validasie	Alles moontlik gedoen om 'n robuuste program te skep, deur toepaslike defensiewe programmeringstegnieke korrek te gebruik, waar nodig.	Goeie gebruik van defensiewe programmeringstegnieke, waar nodig, maar met enkele tekortkominge.	Redelike gebruik van fout-kontrole, met 'n paar basiese foute steeds teenwoordig	Minimale hoeveelheid foutkontrole of defensiewe programmering	Geen poging aangewend nie	4	
Databasis Gebruik en interaksie	4	3	2	1	0		
Inbou van DB manipulasie binne HLL (Hoëvlaktaal) - Kry toegang tot velde as deel van 'n rekord, vir lees/skryf doeleindes en stoor huidige rekord met toepaslike metodes - Navigeer in rekords van 'n datastel, bv. skuif die huidige rekordwyser deur gebruik v. metode of deur indekswaarde te verander - Verander individuele velde of rekords in 'n datastel deur toepaslike metodes en pas alle veranderings toe bv. stoor/redigeer/skrap (save/edit/delete) van huidige rekord of skuif na vorige/volgende rekord	Uitstekende, gladde interaksie met HLL. Goed-gekoose manipulasie deur HLL kode konstruksie wat alles bydrae tot die oplossing. Gebruik ten minste 4 van die gelyste manipulasies	Goeie, gladde interaksie met HLL. Goed-gekoose manipulasie deur HLL kode konstruksie wat alles bydrae tot die oplossing. Gebruik ten minste 3 van die gelyste manipulasies	Aanvaarbare interaksie met HLL. Aanvaarbare manipulasie deur HLL kode konstruksie wat alles bydrae tot die oplossing. Gebruik ten minste 2 van die gelyste manipulasies	Min interaksie met HLL. Min manipulasie deur HLL kode konstruksie wat bydrae tot die oplossing. Gebruik ten minste 1 van die gelyste manipulasies	Geen interaksie, geen manipulasie of geen databasis	4	

• Manipuleer 'n datastel-objek en rekords en pas veranderings toe, bv. filter/soek/sorteer 'n data-stel en verfris die datastel • Gebruik <i>event/listener</i> wat aktiveer nadat rekord-wyser geskuif is -vir validasie van velde -wanneer 'n nuwe rekord geskep, ingevoeg, verander word -om berekenings te doen op nie-tabel velde bv. berekende velde							
'GUI'	4	3	2	1	0		
Gemak van gebruik/HCI-beginsels	Uitstekend – al die volgende: • Baie intuitief • Uitstekende kommunikasie (skerm notas, terugvoer, hulp ens.) • Mees toepaslike komponente • Leesbaar/verstaanbare afvoer • Uitstekende gebruik van effekte/kleure/ikone/kortpaaie (shortcuts), ens.	Goed – een aspek uitgelaat of nie goed genoeg nie	Aanvaarbaar – twee aspekte uitgelaat of nie goed genoeg nie	Beperk – meer as twee aspekte uitgelaat of nie goed genoeg nie	Swak 'GUI'-ontwerp. Min of geen aandag aan HCI-beginsels gegee nie	4	
Dinamiese komponente	Ten minste 2 dinamiese komponente geïnstansieer, albei betekenisvol en korrek gebruik	Ten minste 1 dinamiese komponent geïnstansieer, betekenisvol en korrek gebruik	Dinamiese komponente betekenisvol gebruik, maar werk nie korrek nie	Dinamiese komponente werk korrek, maar is nie betekenisvol nie	Geen dinamiese komponente	4	
Dokumentasie	4	3	2	1	0		
Kommentaar/Notas (Verduideliking van program en kode)	Kode het duidelike notas om alle dele te verduidelik. Verduideliking toon uitstekende insig. Uitgebreide programnotas en van 'n uitstekende standaard. Verduidelik duidelik hoe die program werk	Kode het duidelike notas om alle dele te verduidelik. Verduideliking toon goeie insig. Projeknotas is gebruik en is van baie goeie kwaliteit	Kode het duidelike notas om meeste nodige dele te verduidelik. Verduideliking toon 'n mate van insig. Projeknotas is gebruik en is van gemiddelde kwaliteit.	Kode het notas om sekere dele te verduidelik. Verduideliking toon min insig. Onvoldoende projeknotas is gebruik	Geen kommentaar of projeknotas nie	4	

Algeheel	4	3	2	1	0		
Voldoen die program aan die vereistes?	Oorskry vereistes behoorlik Uitgebreide program, alle elemente funksioneer soos gespesifiseer. Toon in-diepte insig in alle aspekte	Oorskry vereistes Minder uitgebreide program, alle elemente funksioneer soos gespesifiseer. Toon insig in alle aspekte	Geringe oorskryding van vereistes Sommige program elemente funksioneer soos gespesifiseer. Toon insig in 1 of 2 aspekte	Voldoen aan minimum vereistes Basiese program Basiese reikwydte Beperkte insig	Voldoen nie aan minimum vereistes nie Minder as basies Beperkte reikwydte	4	
Aanbieding	2	1	0				
Tydsbestuur – Fase 3	Hou by afsnydatum – alle werk gedoen	Hou nie by afsnydatum nie – 1 dag laat – alle werk gedoen	Hou nie by afsnydatum nie – meer as een dag laat – alle werk nie gedoen nie			2	
Totaal (implementering):						50	

Die volgende tabel bepaal die kompleksiteitsvlak van die program, ten einde te diskrimineer tussen programme op verskillende vlakke. Merk alle eienskappe wat in die program gevind word. **Kompleksiteitsvlak** (Slegs een merkie per reël. Grys blokkies kan nie gemerk word nie. Maal die aantal merkies in elke kolom met die waarde aangedui bo-aan die kolom vir totaal)

Fase 3: Naam van leerder:						
Kompleksiteitsvlak	Kompleks (3)	✓	Voldoende (2)	✓	Beperk (1)	✓
Algoritmes	Nie-triviale algoritmes		Meer gevorderde Grade 11 tipe		Triviale algoritmes	
Gebruiker gedef.	Sluit in veelvuldig geneste lusse/voorwaardelike konstruksie		Slegs dubbel geneste lusse/voorwaardes		Slegs enkel lusse/voorwaardes	
Databasis CRUD bewerkings uitgesluit	Hoë data volumes, baie beperkings, komplekse eie kode Programmeringstaal eie bewerkings vir koppel, stoor/onttrek data Nie-triviale toevoer/afvoer, verwerking, bv. data onttrek moet verder verwerk word om die gewenste resultate te verkry, toeken van navraag parameterwaardes tydens uitvoering, modulêre benadering, bv. gebruik prosedures/funksies/aparte eenhede		Standaard, merkbare data volumes Beperkte eie kode, Slegs standaard bewerkings Standaard toevoer/afvoer, standaard verwerking		Beperk, behels slegs stoor/onttrek en moontlik triviale verwerking met die pakket, feitlik geen eie kode nie Standaard toevoer/afvoer, geen merkbare verwerking nie	
Standaard	Kompleks, nie-triviaal, bv. Fibonacci, faktoriaalfunksie of buite Graad 11 kurrikulum, bv. rekursie		Standaard, veelvuldige bewerkings, binne Graad 11 kurrikulum bv. sorteer, soek, ens.		Eenvoudig, enkel bewerking, gedek in Graad 11 bv. vind van kleinste van twee waardes, ewe onewe, ens.	
Standaard/ gebruiker	Nie-triviale sketse/animasies/grafieke/kaarte/tydsreëling		Standaard sketse/animasies/grafieke/kaarte/tydsreëling		Geen	

Benutting van gesofistikeerde eienskappe van die programmeertaal				
Reikwydte van veranderlikes	Gebruik lokale- en globale-veranderlikes op gepaste wyse en effektief – verbeter die program		Gebruik lokale- en globale-veranderlikes, maar nie altyd op gepaste wyse nie	Beperkte aantal veranderlikes Slegs lokaal
Programmerings- tegnieke	Buite kurrikulum, bv. databasis met meervoudige tabelle (verwantskappe), gebruiksgedefinieerde klasse, 2-D-skikkings, speel videosnitte, 'threads', netwerking, mobiele toeps, tydsgeroeëde simulaties – mag geleende kode wees (Moet korrek werk, toepaslik wees, waarde toevoeg tot oplossing)		Beperkte gebruik van genoemde tegnieke, tegnieke gebruik voeg egter steeds waarde toe tot die oplossing	Geen
Kompleksiteit van nie-rekenaarkonsepte				
Manipulasie van wisk. prosesse:	Insluit van wiskunde bo Graad 12 vlak, bv. 3-D vektor manipulasie Nie-triviale statistiek verskaf		Graad 12 vlak wiskunde Standaard statistiek verskaf, bv. aantal bo gemiddeld, top 10%, ens.	Eenvoudige wiskundige berekenings, bv. optel, aftrek, vermenigvuldiging en deling Statistiek– slegs aggregate soos som, gemiddeld, min
Manipulasie van string prosesse:	Kombineer veelvuldige stringmetodes vir komplekse manipulasies, bv. genereer kode/sleutel, onttrek van dele uit verskeie DB velde/veranderlikes, gebruik 'n kombinasie van stringmetodes		Standaard – Kombineer ten minste twee string metodes	Eenvoudig – slegs enkel manipulasie (gebruik slegs een string metode)
Modulêre aspekte (Hergebruik van kode)				
Hergebruik van kode	Hergebruik van kode – Goeie gebruik van funksies/prosedures/metodes/objekte/parameteroordrag		Goeie gebruik van funksies/prosedures/metodes/objekte/parameteroordrag	Gebruik – ontoepaslike of geen parameteroordrag of werk nie behoorlik nie
Tegniese oplossing				
Uitsonderings/ foutsporing/ validasie	Effektiewe gebruik van toepaslike en effektiewe programmerings-eienskappe en -tegnieke om 'n robuuste oplossing te skep		Goeie gebruik van programmeringseienskappe en -tegnieke om 'n aanvaarbare oplossing te skep	Beperkte gebruik van programmeringseienskappe en -tegnieke om 'n eenvoudige oplossing te skep
	Totaal (Aantal merkies x 3)		Totaal (Aantal merkies x 2)	Totaal (Aantal merkies x 1)
Kolom 1 + Kolom 2 + Kolom 3 (Maksimum: 30) (Kompleksiteit):				Totaal
Finale fase 3 punt: Implementering + Kompleksiteit				

Algemeen: Finale produk en indruk		Leerder naam:				
Aspek	4	3	2	1	0	Mark
Vloei van ontwikkeling	Elke fase van ontwikkeling het logies gevloei uit die vorige fase. Het nie afgewyk van oorspronklike reikwydte ('scope') nie. Die oorspronklike doelwit is bereik en al die vereistes wat in fase 1 genoem is, is bereik.	Moes geringe kwessies en doelwitte uit vorige fases aanspreek. Ten minste 80 % van vereistes is bereik. Sommige aspekte wat oorspronklik beplan was is nie gedoen nie.	Moes 'n aantal kwessies en doelwitte uit vorige fases weer aanspreek. Meer as 50% van die vereistes is nagekom. Sommige aspekte moes verander word of moes opgegradeer of afgeskaal word	Meer as 50% van die oorspronklike vereistes is nie nagekom nie. Baie van die oorspronklike aspekte moes verander, opgegradeer of afgeskaal word.	Feitlik al die oorspronklike vereistes of geen van die oorspronklike vereistes is nagekom nie.	
Professionele projek	Bruikbaar en kan as 'n werklike toepassing geïmplementeer word. Vergelyk goed met gepatenteerde programmatuur. Het professionele eienskappe soos hulpfunksies, goed ontwerpte 'GUI'-uitleg, geen onverwagte foute nie, gebruikersvriendelike terugvoer op alle aspekte en eienskappe.	Kan as 'n werklike toepassing geïmplementeer word met geringe aanpassings. Vergelyk gunstig met gepatenteerde programmatuur. Het omtrent alle professionele eienskappe soos hulpfunksies, goed ontwerpte 'GUI'-uitleg, geen onverwagte foute nie, gebruikersvriendelike terugvoer op alle aspekte en eienskappe.	Kan as 'n werklike toepassing geïmplementeer word met heelwat aanpassings. Vergelyk tot 'n mindere mate met gepatenteerde programmatuur. Het 'n bevredigende aantal professionele eienskappe soos helpfunksies, goed ontwerpte 'GUI'-uitleg, geen onverwagte foute nie, gebruikersvriendelike terugvoer op alle aspekte en eienskappe.	Nie gereed om as 'n werklike toepassing geïmplementeer te word nie. Vergelyk nie goed met gepatenteerde programmatuur nie. Het beperkte professionele eienskappe soos helpfunksies, goed ontwerpte 'GUI'-uitleg, geen onverwagte foute nie, gebruikersvriendelike terugvoer op alle aspekte en eienskappe.	Nie gereed om as 'n werklike toepassing geïmplementeer te word nie. Vergelyk nie met gepatenteerde programmatuur nie. Het feitlik geen professionele eienskappe soos helpfunksies, goed ontwerpte 'GUI'-uitleg, geen onverwagte foute, gebruikersvriendelike terugvoer op alle aspekte en eienskappe.	
Volledigheid	Alle fases van ontwikkeling is volledig en goed ontwerp en uitgevoer. Alle stadiums en fases is goed gedokumenteer	Omtrent alle fases van ontwikkeling is voltooi en goed ontwerp en uitgevoer. Omtrent alle stadiums en fases is goed gedokumenteer	Ten minste twee fases van ontwikkeling is voltooi en goed ontwerp en uitgevoer. Ten minste twee stadiums en fases is goed gedokumenteer	Ten minste een fase van ontwikkeling is voltooi en goed ontwerp en uitgevoer. Ten minste een stadium en fase is goed gedokumenteer	Geen van die fases van ontwikkeling is voltooi en goed ontwerp en uitgevoer nie. Geen van die stadiums en fases is goed gedokumenteer nie	
Houding en toewyding	Alle fases is betyds ingehandig en was goed ontwerp. Werk is gereeld gedoen. Het uitsonderlike toewyding en trots in werk wat tydens elke fase gedoen is, getoon. Het uitsonderlike groei in kennis en vaardighede getoon.	Feitlik alle fases is betyds ingehandig en was goed ontwerp. Werk is gereeld gedoen. Het toewyding en trots in werk wat tydens elke fase gedoen is getoon. Het definitiewe groei in kennis en vaardighede getoon.	Ten minste twee van die fases is betyds ingehandig en was goed ontwerp. Werk is met tussenposes gedoen. Het tot 'n mate toewyding en trots in werk wat gedoen is getoon. Het 'n mate van groei in kennis en vaardighede getoon	Ten minste een van die fases is betyds ingehandig en was goed ontwerp. Werk was nie gereeld gedoen nie. Het beperkte toewyding en trots in werk wat gedoen is getoon. Het beperkte groei in kennis en vaardighede getoon	Geen van die fases is betyds ingehandig en was goed ontwerp nie. Werk was meestal nie gedoen nie. Het geen toewyding en trots in werk wat gedoen is getoon nie. Het geen groei in kennis en vaardighede getoon nie.	
Maksimum: 20						

Aanpassings – %

Mondeling/Onderhoud	100% van finale projekpunt	90% van finale projekpunt	75% van finale projekpunt	60% van finale projekpunt	50% van finale projekpunt	
Verduideliking geselekteerde kode	Verduidelik alle geselekteerde kode duidelik en met selfvertroue Toon uitstekende insig.	Verduidelik alle geselekteerde kode met geringe tekortkominge Toon insig.	Nie in staat om sommige van die geselekteerde kode te verduidelik nie Toon beperkte insig.	Nie in staat om meeste van die geselekteerde kode te verduidelik nie Baie min insig.	Kan geen van die geselekteerde kode verduidelik nie Geen insig.	%
Finale Projekpunt:						

Assesseringsopsomming

Fase	Maksimum Punt	Punt behaal
Fase 1: Analise	30	
Fase 2: Ontwerp	50	
Fase 3: Implementering	50	
Fase 3: Kompleksiteit	30	
Algemeen	20	
Totaal	180	
Aanpassings – %		
Finale punt (Totaal x Aanpassings – %)		

Egtheidsverklaring

Ek verklaar hiermee dat die werk wat geassesseer is slegs die van die leerder is (behalwe waar daar duidelike erkenning gegee is en verslag gelewer is van enige bydraende advies/hulp gegee is aan die leerder) en onder toesig/gekontroleerde toestand gedoen is om te verseker dat die werk nie van iemand anders gekopieer is of voorheen deur enigiemand ingehandig is vir assessering nie.

Kommentaar:

Onderwyser se naam: _____ Onderwyser se handtekening: _____ Datum: _____

Bylaag B

Anatomie van 'n Verslag

Hierdie dokument lig jou in omtrent die spesifieke vereistes van die verslag.

As jy hierdie riglyne volg sal jy verseker wees van 'n suksesvolle verslag. Daar word van jou verwag om 'n stylblad te gebruik sodat jou verslag netjies lyk en ook reg gestruktureer is. Kyk na kolom 1 hieronder vir voorstelle oor formatering. (Jy kan 'n inhoudsopgawe outomaties skep as jy die formatering wat in kolom 1 uitgelig is, gebruik).

TITELBLADSY <title>	Titel van verslag Jou naam en graad Inhandigingsdatum
OPSOMMING <heading>	Oorsig van die verslag se belangrikste inligting en voorstelle
INHOUDSOPGAWE <table of contents>	Lys van genommerde afdelings in die verslag en hulle bladsynommers
INLEIDING <heading 1>	opdrag, die scenario en hoof trekke van die struktuur van die verslag
INHOUD Opskrifte <heading 1> Sub-opskrifte <heading 2>	Opskrifte en subopskrifte wat die inhoud van elke afdeling weergee. Sluit inligting in oor belangrike idee en onderwerpe, en bespreking van programme wat relevant is tot die scenario.
AFSLUITING <heading 1>	Noem die gevolgtrekkings wat uit die inligting wat ingesamel is gemaak kan word, Maak aanbevelings omtrent die rigting wat die leerder moet volg in die projek
LYS VAN VERWYSINGS <heading 1>	Lys met verwysingsmateriaal wat geraadpleeg is tydens navorsing wat vir die projek gedoen is Gebruik die 'Simplified Harvard'-styl/APA-styl
BYLAE <heading 1>	Prente en inligting wat jou navorsing ondersteun maar nie noodsaaklik is in jou verduideliking nie.

Bylaag D**Verklaring van egtheid**

Leerder se naam		ID-nommer	
Graad	12	Jaar	2014
Vak	Inligtingstegnologie		
Praktiese Assesseringstaak (PAT)		Onderwyser	
Ek verklaar hiermee dat die inhoud van hierdie assesseringstaak my eie oorspronklike werk is (behalwe waar daar duidelike erkenning en toepaslike verwysing na die werk van ander is) en dat dit nie onwettig (deur plagiaat) bekom is, van iemand anders gekopieer is, of voorheen vir assessering deur enige persoon ingedien is nie.			
HANDTEKENING VAN LEERDER		____/____/2014 DATUM	

Riglyne vir die Onderwyser

Wat word daar van die leerders verwag om te doen en te lewer?

Daar word van leerders verwag om met toepaslike toesig:

- 'n area van belangstelling binne die onderwerp/scenario te voorsien
- 'n vraag te formuleer waarop gefokus kan word en wat ondersoek/nagevors kan word
- te beplan, na te vors en projek uit te voer
- 'n verslag te lewer vir 'n spesifieke gehoor
- bewys lewer van al die fases van die projek vir assessering.

Hoe sal die leerders te werk gaan?

Die leerders sal:

- 'n individuele projek beplan en voltooi deur 'n verskeidenheid van programmering- en programmatuur-ingeneursvaardighede en -strategieë toe te pas om die doelwitte sooswat dit in die PAT-vereistes uiteengesit is, te bereik
- vrae identifiseer om te vra
- uitgesoekte inligting uit 'n verskeidenheid van bronne bekom, krities uitsoek en gebruik; data verwerk en analiseer, dit binne verband toepas en insig toon met sinvolle aansluiting, verband en kompleksiteit ten opsigte van die onderwerp en die vraag waarop gefokus word
- 'n verskeidenheid van vaardighede kies en gebruik, insluitend ontwerp-gereedskap, algoritmes, probleemoplossing, kritiese besluitneming, kreatiwiteit en buigbaarheid, om 'n programmatuuroplossing te lewer.
- uitkomst evalueer beide in verband met die PAT-vereistes en eie leer en werkverrigting.
- Toepaslike kommunikasievaardighede en media gebruik om bewyse in 'n gepaste formaat aan te bied.

Vaardighede wat vereis word

Die leerder moet in staat wees om:

- navorsing oor die onderwerp te doen en bevindinge behoortlik te dokumenteer insluitend verwysings sooswat gespesifiseer is in Fase 1 se nasienblad.
- 'n volledige analise van gebruikersvereistes te doen wat 'n volledige beskrywing van die rol, aktiwiteite, vereistes en beperkinge van ten minste 3 verskillende gebruikers van die beplande stelsel insluit.
- inligting bymekaar sit om te pas by die inhoud en doel
- besluitneming en probleemoplossingsvaardighede toe te pas
- beplanning, navorsing, kritiese denke, analisering, samestelling, evaluering en voorstellingsvaardighede uitbrei
- selfvertroue te ontwikkel in die toepassing van inhoud, programmering en programmatuur-ingeneurwese-beginsels en tegnieke wat hulle geleer het
- vaardighede skeppend te ontwikkel en toe te pas deur inisiatief en ondernemingsgees te toon
- raad en ondersteuning te soek wanneer nodig.

Wat moet die leerder vooraf geleer word?

Elemente wat geleer is sluit in:

- toepasingsprogrammatuur en IKT-beginsels wat die samestelling van die verslag sal verbeter en die ontwikkeling van die projek wat navorsing, analisering en uitvoer dek
- oplossingsontwikkelingsinhoud en vaardighede insluitend die vermoë om 'n taak te definieer,
- projekbestuursvaardighede insluitend tyd-, bronne- en taakbestuur
- die format en struktuur van aanvaarde vorms van navorsingsverslag wat 'n uittreksel, inleiding, bespreking van al die bronne wat aangehaal is, afsluiting en verwysings

Wanpraktyke

Leerders mag nie:

- hulp/leiding van ander kry sonder om erkenning te gee nie (voltooi **bylae B vir elke fase**)
- ander toelaat om die programmeringkode van hulle projek te doen nie
- werk inhandig wat nie hulle eie is nie
- werk aan ander leerders leen nie
- ander leerders toelaat om toegang te kry tot, of gebruik te maak van hulle eie onafhanklik-verkryde bronmateriaal (dit beteken nie dat kandidate nie hulle boeke vir ander kandidate mag leen nie maar kandidate behoort te keer dat plagiaat met ander leerders se werk gepleeg word)
- werk insluit wat direk uit boeke, die Internet of ander bronne gekopieer is sonder om erkenning te gee en toerekening
- werk wat deur 'n ander persoon getik of verwerk is

Hierdie aksies kom neer op wanpraktyk waarvoor straf toegepas kan word.

As wanpraktyke geïdentifiseer is, moet die assesserende gesag in kennis gestel word en besonderhede van enige werk wat nie die leerder se eie is nie moet aangeteken word.

Leerder-egtheid van die PAT

Vir elke fase voltooi leerders 'n verklaring (**Bylaag B**) vir die werk wat tydens die spesifieke fase gedoen is. Alle selfstandige raad/hulp wat aan leerders gegee is behoort as deel van die fase se dokumentasie aangeteken te word.

Nadat die PAT voltooi is, moet leerders 'n verklaring van egtheid teken (**Bylaag C**) om te bevestig dat die werk wat hulle ingegee het hulle eie is.

Rol van die onderwyser

Die onderwyser sal die inligtingsbestuursinhoud, vaardighede en strategieë onderrig voordat die projek gedoen word.

Die onderwysers sal die projek bestuur en toesig oor die leerders hou:

- hou 'n aanvanklike beplanningsoorsig om die onderwerp/scenario, vereistes, doelwitte en ontwikkeling van die projek te bespreek
- stem saam oor die vraag waarop gefokus moet word (leerders behoort die leiding wat gegee word aan te teken as deel van fase 1 se dokument – **Bylaag B** – bv. waar van toepassing teken eie oorspronklike vraag aan met duidelike bewys van leiding wat gegee en die finale vraag)
- gee gereelde terugvoer aan leerders, bv. om 'n fokus-vraag te formuleer wat geskik en hanteerbaar is
- assesseer die werk van die leerders aan die einde van elke fase deur van die gestandaardiseerde assesseringsinstrument gebruik te maak en terugvoer wat gegee is aan te teken
- bevestig elke leerder se assessering deur die assesseringsinstrument vir elke fase te teken insluitend die finale verklaring dat die bewyse wat ingegee is vir assessering, die werk van die leerder is sonder enige hulpverlening.
- bevestig hulle evaluering op grond van aaneenlopende observasie en terugvoer sowel as 'n voorleggingsessie m 'n finale beoordeling te verskaf omtrent onafhanklike werk, insig en probeemoplossing
- doen die assessering van die leerders se werk deur enige gestandaardiseerde en interne modereringsprosedure wat vereis word te volg

Die onderwyser sal die potensiele projek assesseer (taakdefinisie en wydte) teenoor die volgende kontrolelys.

- Is die fokus-area geskik vir die projek?
- laat die fokus-vraag die leerder toe om toegang tot hoëvlak konsepte en vaardighede in die assesseringsdoelwitte te kry en te ondersoek, dit is beplan, navors, analiseer, evalueer en verduidelik eerder as slegs beskryf en vertel?
- Is die fokus en voorgestelde aksie duidelik en gefokus op 'n kwessie wat bestuur kan word binne die tydgleuf en beskikbare bronne?
- Dui die fokus en voorgestelde aksie aan dat die leerder instaat sal wees om die onderwerp te ondersoek en na te vors en die taak onafhanklik uit te voer binne die geskikte etiese of metodologiese riglyne?
- Sal die leerder dit waarskynlik moeilik vind om die taak en kwessies wat met die fokus geassosieer word te verstaan?

Die onderwyser en die egtheid van die PAT

- Onderwyser moet op die assesseringsinstrument bevestig dat die werk wat geasseseer is alleenlik die betrokke leerder se werk is en dat dit onder toesiggekontroleerde toestande gedoen is.
- Onderwyser moet die assesseringsinstrument van elke fase onderteken

Onder toesig/Gekontroleerde Toestande

Die PAT moet op so 'n manier bestuur word dat dit moontlik is om te kan bevestig dat die werk wat geassesseer is alleenlik die werk van die betrokke leerder is.

Bestuur van die PAT

Die onderwyser moet sy/haar werk skeduleer volgens die tyd wat aan die PAT toegeken is in die IT CAPS (onderrigplan vir Graad 12).

Daar is verskillende moontlike benaderings om die PAT te bestuur:

Opsie 1:

- Die onderwyser kan 'n deel van die tyd op 'n weeklikse basis aan die PAT toewy terwyl daar terselfdertyd voortgegaan word met normale onderrig om die Graad 12 kurrikulum deur die res van die week te voltooi
- As hy/sy hierdie opsie kies behoort hy/sy teen die einde van die eerste kwartaal met die PAT-proses om een fase per kwartaal te voltooi.

Opsie 2:

- Die onderwyser kan 'n aaneenlopende tyd aan die PAT toewy, bv. die laaste week/weke van elke kwartaal en ook een fase per kwartaal voltooi.

Bewys van Assessering

Bewys wat voorgelê is vir assessering moet aandui hoe die individuele leerder aan die assesseringsdoelwitte en kriteria voldoen het en beplanning, terugvoer en vordering met die projek toon.

Die bewys van assessering sal die volgende insluit:

- Die projek as produk insluitend ' geskrewe verslag van ongeveer 1600 woorde (slegs inhoud, uitgesluit die dekblad, inhoudsopgawe, verwsings en grafika), ontwerp-dokumente, finale program (te volle gedokumenteer) en ander bewyse (vir elke fase).
- Die voltooide assesseringsinstrument van die leerder (vir elke fase)

Voorlegging

Riglyne vir die evaluering van die projek:

- Skeduleer datums en tye vir demonstrasies – laat omtrent 15 minute per projek toe.
- Neem al die dokumentasie in voordat die demonstrasie plaasvind – ten minste een week voor die tyd – en evalueer die dokumentasie voor die demonstrasie-sessie.
- Leerders behoort hulle projekte op die rekenaar te demonstreer.
- Tydens die demonstrasie-sessie behoort leerders toets-prosedures uit te voer om te wys dat die hele program reg werk.

- Gebruik die nasienblad vir Fase 3 as 'n riglyn en ken punte daarvolgens toe tydens die demonstrasie.
- Identifiseer stukke programmeringskode in die projek ewekansig as deel van die evaluering en vra die leerder om die doel en werking van die ewekansig-geselekteerde kode te verduidelik. Dit word gedoen om te verseker dat die leerder self die kode gedoen het. 'n Soortgelyke tipe prosedure sal gevolg word tydens moderering. As 'n leerder nie die kode in die projek kan verduidelik nie sal 'n punt van nul aan die projek toegeken word.
- Maak seker dat die leerder die elektroniese kopie van die projek wat gedemonstreer was inhandig. Gebruik hierdie kopie om enige uitstaande punte toe te ken om die punt te finaliseer.

Vereistes

(Nasionale Protokol vir Assessering Graad R – 12, Hoofstuk 3)

Komponente van die Praktiese Assesseringstaak moet:

- bestaan uit assesseringstake saam deel uitmaak van die leerder se PAT-punt soos beoog is in Hoofstuk 4 van die Kurrikulum en Assesseringsbeleidstelling vir IT
- 'n punt insluit wat toegeken is vir elke assesseringstaak (fase) en 'n konsoliderende punt
- gelei word deur die assesseringskomponent soos gespesifiseer in Hoofstuk 4 van die Kurrikulum en Assesseringsbeleidstelling vir IT
- beskikbaar wees vir monitering en moderering
- ge-evalueer, gekontroleer en eg bevind is deur die onderwyser voordat dit voorgelê word as die leerder se bewys van werkverrigting

Nie nagekom nie

(Nasionale Protokol vir Assessering Graad R – 12, Hoofstuk 3)

Die afwesigheid van 'n PAT-punt in IT sonder 'n geldige rede sal uitloop daarop dat die leerder nie bepunt word nie.

Die leerders sal tot drie weke voor die aanvang van die finale einde-van-die-jaar-eksamen gegee word om uitstaande werk in te handig of om hom-/haarself voor te stel vir die PAT. As die leerder misluk daarin om te voldoen aan die uitstaande vereistes van die PAT, nul ('0') aan die leerder toegeken word aan die PAT-komponent vir IT.

In die geval van 'n leerder wat nie voldoen aan die vereistes van die PT nie maar waar daar 'n geldige rede gegee is:

- Hy of sy sal 'n ander geleentheid gegee word om geassesseer word in take wat toegeken is wat gebaseer op 'n besluit wat deur die Hoof van die assesseingliggaam geneem is.
- Die leerder moet binne drie weke voor die aanvang van die finale einde-van-die-jaar-eksamen die uitstaande werk inhandig of sy/haar PAT voorlê.
- Indien die leerder nie daarn slaag om aan die uitstaande vereistes van die PT te voldoen nie sal die punt van die PAT-komponent weggelaat word en die finale punt sal aangepas word vir promosiedoeleindes met betrekking tot die voltooide take.

Geldige redes in hierdie konteks sluit die volgende in:

- Siekte wat bevestig word deur 'n geldige mediese sertifikaat wat deur 'n geregistreerde mediese praktisyn uitgereik is
- Menslikheidsredes wat die dood van 'n onmiddelijke familielid wat bevestig is deur 'n doodsertifikaat insluit
- Die leerder in 'n hofsaak moet verskyn wat bevestig moet word deur geskrewe bewys
- Enige ander rede wat as geldig aanvaar kan word deur die Hoof van die assesseringsliggaam of sy/haar verteenwoordiger

As die leerder nie aan die vereistes van die Praktiese Assesseringstaak van 'n spesifieke vak voldoen nie en waar daar geldige redes verskaf is, moet die bewyse van die geldige redes ingesluit wees by die bewyse van die leerder se werkswaardigheid.