

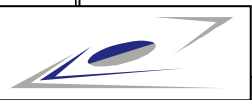
**ИСПИТНА
ПРОГРАМА**



**МИНИСТЕРСТВО ЗА ОБРАЗОВАНИЕ И НАУКА
БИРО ЗА РАЗВОЈ НА ОБРАЗОВАНИЕТО
СЕКТОР ЗА ИСПИТИ**

ПРОГРАМИРАЊЕ

ДРЖАВНА МАТУРА И ЗАВРШЕН ИСПИТ



**СРЕДНО СТРУЧНО
ОБРАЗОВАНИЕ**

Скопје 2005

МИНИСТЕРСТВО ЗА ОБРАЗОВАНИЕ И НАУКА
БИРО ЗА РАЗВОЈ НА ОБРАЗОВАНИЕТО
СЕКТОР ЗА ИСПИТИ

ИСПИТНА ПРОГРАМА

ПРОГРАМИРАЊЕ

ДРЖАВНА МАТУРА И ЗАВРШЕН ИСПИТ

Испитната програма по *програмирање* за државна матура и за завршен испит е донесена со решение на министерот за образование и наука бр. 11-136/1 од 11.01.2006 година.

1. ВОВЕД

Со програмата за реформа на средното стручно образование и обука во Република Македонија (која почна да се реализира од учебната 1999/2000 година) и новиот наставен план и програми за гимназиското образование (кој започна во учебната 2001/2002 година) воведени се новини кои резултираа со промени во начинот на следење и оценување на постигањата на учениците и дефинирање Концепција за матура и за завршен испит во јавното средно образование.

Според Концепцијата, основната цел на овие испити е:

- подигање на квалитетот на средното образование;
- следење и контрола на реализацијата на наставните програми (засновани на образовните стандарди на државно ниво);
- стекнување диплома за завршено средно образование (заокружување на образованието со испит);
- селекција за универзитетското образование;
- информирање на учениците, родителите и образовните институции за постигањата на учениците добиени преку валидни и веродостојни вреднувања.

Според целите и содржините, испитната програма за државна матура и за завршен испит по предметот програмирање се базира врз наставните програми за II, III и IV година по предметот програмирање кој се изучува во електротехничката струка за образовниот профил *електротехничар за компјутерска техника и автоматика* во средното стручно образование.

Предметот програмирање спаѓа во изборниот дел од матурскиот испит и е во листата на наставни предмети за третиот предмет од изборниот дел на државната матура и листата на наставни предмети за полагање на изборниот дел од завршниот испит во стручното образование што се реализира според нови наставни планови и програми.

Испитната програма ги содржи следните компоненти:

- Општа цел на испитот
- Содржина на испитот
- Спецификација на подрачјата и способностите
- Конкретизација на целите
- Спецификациска мрежа на испитот
- Опис на испитот
- Начин на оценување.

2. ОПШТА ЦЕЛ НА ИСПИТОТ

Целта на матурскиот испит по програмирање е да се провери:

- колкаво е нивото на способностите кои ги стекнал ученикот и неговата подготвеност за успешно да се вклучи во процесот на работата;
- колку е оспособен да ги користи знаењата по програмирање за решавање на проблеми со помош на компјутер;
- дали има стекнато знаења и способности кои ќе му овозможат успешно да го продолжи своето образование во високообразовните институции;
- колку стекнал знаења за основните поими од програмирање (паскал, турбопаскал, компајлери и интерпретери);
- колку стекнал знаења за едноставни типови на податоци;
- колку стекнал знаења за сложени типови на податоци;
- колку ученикот стекнал знаења за визуелен програмски пакет - делфи.

За успешно полагање на училишниот матурски испит по програмирање, ученикот треба:

- да препознава и создава програмски код од програмскиот јазик ПАСКАЛ;
- да разбира основни поими од програмирање (паскал, турбопаскал, компајлери и интерпретери);
- да препознава, користи и решава задачи со едноставни типови на податоци;
- да препознава, користи и решава задачи со сложени типови на податоци;
- да решава проблеми со помош на програмскиот јазик ПАСКАЛ;
- да проценува и одредува методи и начини најсоодветни за решавање на дадени проблеми со помош на програмскиот јазик ПАСКАЛ;
- да решава проблеми со делфи (прави претпоставки, анализира, проценува, избира ефикасен циклус, тип на податок, прави соодветен модел, толкува избран циклус и условен и безусловен премин, применува знаење на поими и постапки, организира и толкува информации, решава нестандартни проблеми и образложува - докажува како и проверува коректност на избрана постапка или модел за решавање и вреднува точност-разумност на добиено решение);
- да ја разбира меѓусебната поврзаност на различните подрачја од науката и техниката во програмирањето.

3. СОДРЖИНА НА ИСПИТОТ

3.1. Спецификација на подрачјата (содржините) и способностите

Во испитната програма се опфатени подрачјата:

- **ЕДНОСТАВНИ ТИПОВИ НА ПОДАТОЦИ**
- **СЛОЖЕНИ ТИПОВИ НА ПОДАТОЦИ И ПОИНТЕРИ**
- **ВИЗУЕЛЕН ПРОГРАМСКИ ПАКЕТ - ДЕЛПХИ**

Подолу групирано се дадени способностите кои ученикот треба да ги поседува за успешно решавање на испитните задачи:

- **C1- препознавање и разбирање (идентификување, именување, наведување, заокружување, поврзување, подвлекување, регистрирање, набројување);**
- **C2- примена (разликување, поврзување, опишување, објаснување, наведување примери, дополнување, конструирање);**
- **C3- решавање задачи (класифицирање, категоризирање, развивање, ревидирање, организирање, подредување, споредување).**

3.2. Конкретизација на целите (знаењата и способностите) по подрачја

ПОДРАЧЈЕ 1: ЕДНОСТАВНИ ТИПОВИ НА ПОДАТОЦИ	
Содржина	Знаења и способности
<ul style="list-style-type: none"> - Поим за програмирање - Поим за податок - Поим за компајлер и интерпретер, како и нивни сличности и разлики 	<p>Ученикот треба:</p> <ul style="list-style-type: none"> - да објасни што е програмирање на компјутер; - да објасни што е програмирање; - да објасни како се остварува програмирањето, односно како се добива извршен (exe) код и како се дистрибуира готовата програма.
<ul style="list-style-type: none"> - Целоброен тип на податок INTEGER - Реален тип на податок REAL - Знаковен тип на податок CHAR - Низовен тип на податок string - Логичен тип на податок boolean - Аритметички, логички и релациони оператори и нивниот приоритет - Основен модел на паскал програма (наслов, коментар и наредби) 	<p>Ученикот треба:</p> <ul style="list-style-type: none"> - да објасни што се тоа едноставни типови на податоци; - да препознава едноставни типови на податоци; - да декларира едноставни типови на податоци; - да употребува во даден момент соодветен тип на податок; - да креира променливи; - да креира константи; - да декларира нови типови на податоци; - да применува соодветно аритметички, логички и релациони оператори во даден момент; - да познава приоритет на аритметички, логички и релациони оператори; - да познава основен модел на една паскал програма: наслов на програма, коментар и наредби; - да објасни што е тоа коментар, кога, каде и како се користи; - да објасни за поделба на наредби на извршни и неизвршни и како се користат; - да препознава израз со повеќе оператори, со загради и да го користи приоритетот на операторите според правилото од лево на десно; - да објаснува за циклуси: кои се и кога односно во кој момент се користат; - да објаснува за условни гранења и кое да го употреби во даден момент; - да решава задачи (прави програми) со употреба на едноставни типови на податоци, циклуси и условни гранења;

<ul style="list-style-type: none"> - Изрази со еден оператор - Изрази со повеќе оператори - Изрази со загради и нивното влијание на приоритетите на операторите - Декларирање на променливи - Декларирање на константи - Декларирање на нови типови на податоци врз основа на постоечките - Циклуси: поим и видови (while, for, repeat-until) - Условни гранења (if и case) - Основни команди за прием на податоци од тастатура, како и приказ на резултатите од обработката на екран или печататар (write, writeln, read, readln) 	<ul style="list-style-type: none"> - да препознава решенија од дадена програма, односно анализа на програмскиот код од аспект кој ред што работи и/или каков резултат ќе испечати кодот; - да прими податок од тастатура; - да испечати податок-резултат на екран; - да форматира (обликува) излезен податок; - да препознава изрази со повеќе оператори и/или со загради.
--	---

ПОДРАЧЈЕ 2: СЛОЖЕНИ ТИПОВИ НА ПОДАТОЦИ И ПОИНТЕРИ

Содржина	- Знаења и способности
<ul style="list-style-type: none">- Еднодимензионални полиња - низи arrayП - пакувани полиња packed array- Дводимензионални полиња matrix- Поинтери - pointer- Записи - records- Датотеки - files- Множества - sets- Процедури - procedure- Функции - function	<p>Ученикот треба:</p> <ul style="list-style-type: none">- да објасни што се тоа сложени типови на податоци;- да препознава и декларира еднодимензионални полиња (низ);- да примени постапка за полнење на една низа;- да изврши аритметички и/или логички операции врз податоците од низата;- да презентира резултати од таква обработка;- да објасни што се тоа пакувани полиња, како се декларираат и како и каде се користат;- да решава задачи каде што се користат низите;- да анализира даден код со низи и да го протолкува кодот како и резултатите од обработката на податоците во низата;- да препознава и декларира дводимензионални полиња (матрици);- да примени постапка за полнење на една матрица;- да изврши аритметички и/или логички операции врз податоците од матрицата;- да презентира резултати од таква обработка;- да решава задачи каде се користат матриците;- да анализира даден код со матрици и да го протолкува кодот како и резултатите од обработката на податоците во матрицата;- да објасни што се поинтери, кога и како се употребуваат - најчесто во склоп на обработката на податоците од записите;- да препознава и објасни за запис, разлики и сличности со низите и матриците;- да декларира запис, да полни запис, да извршува аритметички и логички операции врз податоците од записите(со поинтери и/или без нив);- да решава задачи каде се користат записите;- да анализира даден код со записи и да го протолкува кодот како и резултатите од обработката на податоците во записот;- да користи матрични записи како основа на базите на податоци;- да презентира резултатите од обработката на податоците во записи(со поинтери и/или без нив);- да објасни за датотека: што се тие, како ги запишуваме, како ги препознаваме,

- какви датотеки постојат и со какви датотеки работи програмскиот јазик паскал;
- да ја користи командата за доделување на датотека **асигн**;
 - да ги познава неколкуте стандардни процедури за вчитување и читање на датотеки. Тоа се процедурите: **RESET, REWRITE, GET, PUT**;
 - да знае дека текстуалните датотеки многу често се користат. во паскал и дека паскалот дава додатни стандардни процедури за работа со нив. Тоа се: **read, readln, write, writeln**;
 - да решава задачи каде се користат датотеки;
 - да анализира даден код со датотеки и да го протолкува кодот како и резултатите од обработката на податоците во датотеката;
 - да препознава и декларира множества;
 - да објаснува основни операции со множества: унија, пресек и разлика;
 - да применува релациони оператори наменети за множества;
 - да решава задачи со множества;
 - да објасни што се тоа процедури и функции: како се користат, кога се користат и каде се користат и разликите помеѓу нив;
 - да употребува функции;
 - да пишува функции за проблеми што се повторуваат;
 - да употребува процедури;
 - да пишува процедури за одделни проблеми.

ПОДРАЧЈЕ 3: ВИЗУЕЛЕН ПРОГРАМСКИ ПАКЕТ ДЕЛПХИ	
Содржина	Знаења и способности
<p>ВОВЕД ВО ВИЗУЕЛНО ПРОГРАМИРАЊЕ</p> <ul style="list-style-type: none"> - WINDOWS работна околина - Поим за објектно ориентирано програмирање 	<p>Ученикот треба:</p> <ul style="list-style-type: none"> - да препознава што е тоа ЊИНДОЊС работна околина; - да разликува DOS и windows оперативен систем; - да објаснува поими single user, multi user, multitasking; - да разбира што значи објектно ориентирано програмирање.
<p>ПРОГРАМСКИ ПАКЕТ ДЕЛФИ</p> <ul style="list-style-type: none"> - Делови на делфи екранот - Работа со главниот прозорец - Опции на главното мени - Палета на компоненти - Најчесто користени компонент; - Видови датотеки во еден проект - Работа со код - Формата како главна компонента - Нанесување на компоненти на формата и промена на нивните својства (рачно и/или програмски) 	<p>Ученикот треба:</p> <ul style="list-style-type: none"> - да ги препознава 4-те делови на делфи екранот; - да применува object inspektor и картички properties и events; - да препознава главно мени, главна (почетна) форма; - да препознава палета на компоненти; - да препознава лента со икони и нивна организација; - да користи поважни опции од главно мени и нивни подопции; <ul style="list-style-type: none"> - FILE (new, new aplication, new form, open, open project, reopen, save, save as, save all, save project as, close, close all, print, exit); - EDIT (undo, redo, cut, copy, past, select, align to grid, bring to front, align, size, scale, tab order i lock controls); - VIEW; - PROJECT; - RUN; - COMPONENT; - DATABASE; - TOOLS ; - HELP; - да препознава, селектира и користи компоненти во даден момент најчесто користени:

<ul style="list-style-type: none"> - Декларирање и употреба на променливи и константи; - Типови на константи и променливи во делфи; - Контрола на текот на извршувањето на кодот(if и case); - Циклуси (for,while и repeat - until); - Функции за креирање на дијалог рамки; - Компонента за правење на менија; - Користење на мрежа за приказ на податоци (grid); - Процедури и функции во делфи; - Функции (цел број во стринг, стринг во цел број, реален во стринг и обратно); - Бази на податоци (поле, запис, табела); - Поделба на базите на податоци; - Основни својства, методи и настани кај компонентите за бази на податоци 	<ul style="list-style-type: none"> - (tlabel, tedit, ttimer, tbutton); - да селектира одреден настан и генерира код за истиот за соодветна компонента; - да логички и креативно организира уредување на форма како главна и прва компонента; - да објаснува какви датотеки постојат во еден проект; - да променува својства и настани програмски или рачно на некоја компонента; - да анализира процедури и функции во делфи; - да синтетизира процедури и функции во делфи; - да креира во даден момент дијалог рамка (showmessage, messagedlg i messagedlgpos); - да креира менија за дадена апликација (проект); - да препознава поле, запис, табела и база на податоци; - да препознава бази на податоци; - да разликува бази на податоци; - да користи компоненти за бази на податоци; - да разбира својства, методи и настани од компонентите за работа со бази на податоци; - да познава методи за движење низ табела (first, last, next prior, bof, eof, moveby(n)); - да ја дефинира состојбата на датасетот (insert, append, edit, open, close); - да синтетизира програми со компоненти за пристап кон бази на податоци (ttable, tquery, tdatasource) како и компоненти за приказ на податоци од истите (tdbedit, tdbgrid, tdbtext, tdbnavigator, tdbmemo); - да креира програмски код за полнење на табели, пребарување на табели (setkey, gotokey).
---	---

4. СПЕЦИФИКАЦИСКА МРЕЖА НА ИСПИТОТ

Во следнава шема е дадена процентуалната застапеност на подрачјата и способностите во тестот по програмирање. Бројот на испитните задачи од секое подрачје, кои вклучуваат и одредена група способности, ќе биде соодветен на нивната процентуална застапеност во однос на вкупниот број испитни задачи кои ќе ги содржи тестот.

СПОСОБНОСТИ	ПОДРАЧЈА			ЗАСТАПЕНОСТ (%)
	П1	П2	П3	
С1				20-40
С2				20-40
С3				20-50
ЗАСТАПЕНОСТ (%)	10-30	20-50	20-50	100 %

С1 - познава поими, препознава, разбира, дефинира, декларира, определува

С2 - анализира, размислува, одлучува, идентификува, споредува, опишува

С3 - анализира, синтетизира, логички размислува, предлага, одлучува, составува

П1 - ЕДНОСТАВНИ ТИПОВИ НА ПОДАТОЦИ

П2 - СЛОЖЕНИ ТИПОВИ НА ПОДАТОЦИ И ПОИНТЕРИ

П3 - ВИЗУЕЛЕН ПРОГРАМСКИ ПАКЕТ - ДЕЛПХИ

5. ОПИС НА ИСПИТОТ

Испитот по предметот програмирање е писмен.

Испитот се состои во решавање тест.

Времетраењето на испитот по програмирање е 120 минути и се спроведува без пауза.

Тестот содржи околу 45 испитни задачи.

Во тестот се застапени испитни задачи од следните видови:

- задачи во кои ученикот треба да избере еден точен од повеќе понудени одговори;
- отворени задачи - задачи во кои треба на означеното место да запише одговор;
- задачи во кои ученикот треба да ја покаже целата постапка на решавање.

За време на испитот ученикот може да користи калкулатор.

6. НАЧИН НА ОЦЕНУВАЊЕ

Вкупниот број бодови што можат да се освојат на испитот е околу **100**.

Точниот одговор на задачите со повеќечлен избор (во кои се бара ученикот да избере еден од одговорите што се понудени) се оценува со 1 бод. Доколку точно ги реши сите задачи од овој тип, ученикот може да освои околу **15 бода**.

Точниот одговор во задачите на кои се бара директен одговор (со еден или неколку зборови), се оценува со 1 - 3 бода. Со точното решавање на ваквите задачи ученикот може да освои околу **50 бода**.

Задачите на кои се бара да се покаже како се решава некоја задача (проблемска ситуација), да се дискутира, образложи или вреднува некое решение или став, се оценуваат така што одделно се оценува точното решавање во секоја фаза од одговарањето на барањата на задачата. Зависно од бројот на барањата овие задачи се оценуваат со повеќе од 3 бода. На ваквите задачи, ако точно ги реши, ученикот може да освои околу **35 бода**.

Оценувањето ќе се врши интерно, од страна на училишната предметна комисија, а врз основа на однапред изготвено упатство и критериуми.