

Programiranje I :: Priprema za integralni ispit

Zadatak 1

Napisati program za unos n tekstova (gdje je n ukupan broj tekstova koji unosi korisnik). Svaki tekst mora biti kreiran kao dinamički niz karaktera. Zatim, po mogućnosti možete koristiti funkcije sa sljedećim prototipom:

Funkcija	Ulazi	Izlaz	Definicija
<i>count</i>	<i>char ** tekstovi,</i> <i>int brojTekstova,</i> <i>const char * rijec</i>	<i>int</i>	Prebrojava sve tekstove u nizu koji sadrže proslijeđenu vrijednost <i>rijec</i> i vraća ukupan broj istih.
<i>isSingleWord</i>	<i>const char * tekst</i>	<i>bool</i>	Funkcija za provjeru da li je proslijeđeni tekst zapravo jedna riječ.

Nakon korisničkog unosa tekstova (koji mogu biti različitih dužina), potrebno je omogućiti unos jedne riječi (statički niz karaktera veličine 50), uz napomenu da se unos ponavlja dokle god nije unesena isključivo jedna riječ.

Nakon uspješnog unosa jedne riječi, potrebno je pozvati funkciju *count* i ispisati ukupan broj tekstova u kojima možemo pronaći unesenu riječ.

Napomena: Zabranjeno je korištenje klase *string* i dealokacija svih tekstova je obavezna.

Microsoft Visual Studio Debug Console

```
Unesite broj tekstova:
5
Unesite tekst na indeksu 0:
Indy Neidell
Unesite tekst na indeksu 1:
Indiana (Indy) Jones
Unesite tekst na indeksu 2:
Han Solo
Unesite tekst na indeksu 3:
Luke Skywalker
Unesite tekst na indeksu 4:
Josep Pep Guardiola
Tekst 0: Indy Neidell
Tekst 1: Indiana (Indy) Jones
Tekst 2: Han Solo
Tekst 3: Luke Skywalker
Tekst 4: Josep Pep Guardiola
Unesite jednu rijec:
Mikel Arteta
Unesite jednu rijec:
Indy
Broj tekstova u kojima mozemo pronaci rijec je 2

C:\Users\kema1\source\repos\TestingGround\Debug\TestingGround.exe (process 11636) exited with code 0.
Press any key to close this window . . .
```



Zadatak 2

Napišite program koji omogućuje korisniku unos teksta (preko dinamičke alokacije) i zatim pronalazi najdužu riječ u unesenom tekstu. Dužina riječi se može odrediti brojem slova (alfabetskih simbola), a riječi se međusobno odvajaju bilo kojim znakom koji nije slovo (npr. razmak, interpunkcijski znak ili slično). Program treba ispisati dužinu najduže riječi i samu riječ.

Napomena: Zabranjeno je korištenje klase *string* i dealokacija niza karaktera je obavezna.

```

Microsoft Visual Studio Debug Console
Unesite tekst:
matrix#space odyssey#1984 fahrenheit 451
Najveca duzina je: 10
Najduza rijec je: fahrenheit

C:\Users\kemal\source\repos\TestingGround\Debug\TestingGround.exe (process 22392) exited with code 0.
Press any key to close this window . . .

```

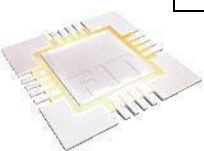
Zadatak 3

Napišite program koji omogućuje unos jednodimenzionalnog niza cjelobrojnih vrijednosti preko niza pokazivača (`int*`). Uraditi neophodne alokacije za svaki pokazivač u nizu a zatim obavezno implementirati sljedeće funkcije:

Funkcija	Ulazi	Izlaz	Definicija
<i>enterArray</i>	<i>int ** parray,</i> <i>int i,</i> <i>int max</i>	<i>void</i>	Rekurzivna funkcija zadužena za unos vrijednosti elemenata niza. Parametar <i>parray</i> odnosi se na niz pokazivača, dok se parametar <i>max</i> odnosi na veličinu niza.
<i>printArray</i>	<i>int ** parray,</i> <i>int i,</i> <i>int max</i>	<i>void</i>	Rekurzivna funkcija zadužena za ispis vrijednosti elemenata niza. Parametar <i>parray</i> odnosi se na niz pokazivača, dok se parametar <i>max</i> odnosi na veličinu niza.
<i>sumArray</i>	<i>int ** parray,</i> <i>int i,</i> <i>int max</i>	<i>int</i>	Rekurzivna funkcija zadužena za sumiranje vrijednosti elemenata niza. Parametar <i>parray</i> odnosi se na niz pokazivača, dok se parametar <i>max</i> odnosi na veličinu niza.
<i>findElement</i>	<i>int ** parray,</i> <i>int i,</i> <i>int max</i> <i>int target</i>	<i>int</i>	Rekurzivna funkcija zadužena za pronalaženje vrijednosti <i>target</i> u nizu. Parametar <i>parray</i> odnosi se na niz pokazivača, dok se parametar <i>max</i> odnosi na veličinu niza. Ukoliko je vrijednost <i>target</i> pronađena u nizu, potrebno je vratiti indeks odgovarajućeg elementa; u suprotnom vratiti vrijednost -1.

U main funkciji dodati sljedeće stavke:

- Kreiranje statičkog niza pokazivača veličine 10
- Alokacija svakog pokazivača
- Unos elemenata preko tastature (poziv rekurzivne funkcije)



- Ispis svih elemenata (poziv rekurzivne funkcije)
- Ispis sume svih elemenata (poziv rekurzivne funkcije)
- Unos vrijednosti *target* za pretragu te poziv odgovarajuće (rekurzivne funkcije)
- Dealokacija

```

Microsoft Visual Studio Debug Console
Unesite element na lokaciji 0:
11
Unesite element na lokaciji 1:
22
Unesite element na lokaciji 2:
33
Unesite element na lokaciji 3:
44
Unesite element na lokaciji 4:
55
Unesite element na lokaciji 5:
66
Unesite element na lokaciji 6:
77
Unesite element na lokaciji 7:
88
Unesite element na lokaciji 8:
99
Unesite element na lokaciji 9:
11
Ispis:
11 22 33 44 55 66 77 88 99 11
Suma elemenata: 506

Unesite element za pretragu: 44
Element 44 pronadjen na indeksu 3

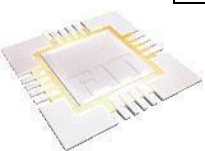
C:\Users\kemal\source\repos\TestingGround\Debug\TestingGround.exe (process 4320) exited with code 0.
Press any key to close this window . . .

```

Zadatak 4

Napišite program gdje je potrebno uraditi implementaciju navedenih funkcija:

Funkcija	Ulazi	Izlaz	Definicija
<i>getRandomValue</i>	<i>int from</i> <i>int to</i>	<i>int</i>	Funkcija koja vraća random vrijednost između dvije proslijeđene vrijednosti.
<i>createMatrix</i>	<i>int rows</i> <i>int columns</i>	<i>int **</i>	Funkcija koja kreira (dinamički alocira) matricu dimenzija <i>rows</i> × <i>columns</i> .
<i>setRandomValues</i>	<i>int ** matrix</i> <i>int rows</i> <i>int columns</i> <i>int from</i> <i>int to</i>	<i>void</i>	Funkcija koja radi postavljanje svih elemenata matrice na slučajne vrijednosti u opsegu koji je naveden vrijednostima <i>from</i> i <i>to</i> .
<i>printMatrix</i>	<i>int ** matrix</i> <i>int rows</i> <i>int columns</i>	<i>void</i>	Funkcija koja radi ispis elemenata matrice. Obavezno koristiti <i>setw</i> (iz <i>iomanip</i> biblioteke) prilikom formatiranja ispisa.
<i>diagonalStack</i>	<i>int ** m1</i> <i>int ** m2</i> <i>int r1</i> <i>int c1</i>	<i>int **</i>	Funkcija zadužena za dijagonalno slaganje matrica <i>m1</i> i <i>m2</i> . U ovom kontekstu, primjer dijagonalnog slaganja je prikazan na jednoj od



	<i>int r2</i> <i>int c2</i>		ilustracija ispod. Rezultat funkcije je nova matrica.
<i>deleteMatrix</i>	<i>int ** matrix</i> <i>int rows</i>	<i>void</i>	Funkcija zadužena za dealokaciju matrice.

Neka matrica *m1* ima sljedeće elemente:

```
1 2 3
4 5 6
7 8 9
```

i neka matrica *m2* ima sljedeće elemente:

```
1 1
2 2
3 3
```

Rezultujuća matrica (matrica nastala dijagonalnim slaganjem) će imati sljedeći oblik:

```
1 2 3 0 0
4 5 6 0 0
7 8 9 0 0
0 0 0 1 1
0 0 0 2 2
0 0 0 3 3
```

U main funkciji dodati sljedeće stavke:

- Unos broja redova i kolona za prvu matricu *m1* (ponavljati unos ako su unesene nevalidne vrijednosti)
- Unos broja redova i kolona za drugu matricu *m2* (ponavljati unos ako su unesene nevalidne vrijednosti)
- Kreiranje matrica *m1* i *m2*.
- Randomizacija vrijednosti matrica *m1* i *m2* vrijednostima između 1 i 10.
- Ispis matrica *m1* i *m2*.
- Kreiranje matrice *m3* pomoću dijagonalnog slaganja matrica *m1* i *m2*.
- Ispisivanje matrice *m3*.
- Dealokacija

Napomena: Dealokacija svih korištenih matrica je obavezna.

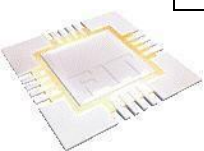
```
Microsoft Visual Studio Debug Console

Unesite broj redova za prvu matricu:
3
Unesite broj kolona za prvu matricu:
3
Unesite broj redova za drugu matricu:
3
Unesite broj kolona za drugu matricu:
2
Matrica M1:
6 1 10
7 4 10
7 5 10

Matrica M2:
10 10
1 2
10 7

Matrica M3 formirana dijagonalnim slaganjem matrica M1 i M2:
6 1 10 0 0
7 4 10 0 0
7 5 10 0 0
0 0 0 10 10
0 0 0 1 2
0 0 0 10 7

C:\Users\kemal\source\repos\TestingGround\Debug\TestingGround.exe (process 12420) exited with code 0.
Press any key to close this window . . .
```



Zadatak 5

Napišite program gdje je potrebno uraditi implementaciju navedene funkcije:

Funkcija	Ulazi	Izlaz	Definicija
<i>rotateArray</i>	<pre>int ** parray int size int rotateWindow</pre>	<i>void</i>	<p>Funkcija zadužena za rotaciju elemenata niza pokazivača za <i>rotateWindow</i> koraka ulijevo.</p> <p>Npr. neka dinamičke varijable (na koje pokazuju pokazivači iz niza) imaju sljedeće vrijednosti: [11 22 33 44 55 66 77] i neka je <i>rotateWindow</i> = 2 tada je rezultujući niz imati vrijednosti: 33 44 55 66 77 11 22]</p> <p>Vrijednost <i>rotateWindow</i> ne smije biti veća od veličine dinamičkog niza <i>size</i>.</p>

U main funkciji dodati sljedeće stavke:

- Unos veličine dinamičkog niza pokazivača *vel* (uz ograničenje *vel* > 0)
- Alokacija dinamičkog niza pokazivača
- Alokacija svakog pokazivača iz niza
- Unos elemenata
- Ispis elemenata
- Unos vrijednosti *rotationWindow* (uz ograničenje $0 < rotationWindow < vel$)
- Rotacija niza pomoću poziva funkcije *rotateArray*
- Ispis elemenata
- Dealokacija pokazivača iz niza
- Dealokacija dinamičkog niza

Napomena: Obavezno koristiti dinamički (a ne statički) niz pokazivača. Dealokacije (kako pojedinačnih pokazivača iz niza tako i dinamičkog niza) su također obavezne.

```

Microsoft Visual Studio Debug Console

Unesite velicinu dinamickog niza pokazivaca:
7
Unesite element na lokaciji 0: 11
Unesite element na lokaciji 1: 22
Unesite element na lokaciji 2: 33
Unesite element na lokaciji 3: 44
Unesite element na lokaciji 4: 55
Unesite element na lokaciji 5: 66
Unesite element na lokaciji 6: 77
Ispis elemenata niza [PRIJE ROTACIJE]
11 22 33 44 55 66 77
Unesite velicinu rotacijskog prozora (broj skokova ulijevo):
2
Ispis elemenata niza [POSLIJE ROTACIJE]
33 44 55 66 77 11 22

C:\Users\kemal\source\repos\TestingGround\Debug\TestingGround.exe (process 708) exited with code 0.
Press any key to close this window . . .

```



Zadatak 6

Napišite program u kojem je potrebno (u main funkciji) implementirati sljedeće stavke:

- Kreiranje niza tekstova veličine 10 (statički niz pokazivača tipa *char*)
- Unos tekstova (različitih dužina) preko tastature i adekvatna dinamička alokacija memorije za pohranu istih
- Kreiranje novog teksta (koji je dinamički alociran) u koji će se pohraniti rezultat konkatencije tj. procesa spajanja svih tekstova iz niza. Spajanje se mora uraditi na način da se između susjednih tekstova mora naći znak za prazan prostor (engl. empty space). Voditi računa o veličini dinamičke memorije uzimajući u obzir dužine tekstova iz niza kao i prazne prostore.
- Nakon što se svi tekstovi uspješno dodaju u jedan zajednički tekst, uraditi „kapitalizaciju“ početnih slova za sve riječi u dobijenom tekstu.
- Ispisati rezultujući tekst

Napomena: Zabranjeno je korištenje klase *string* i dealokacija svakog teksta je obavezna.



```

Microsoft Visual Studio Debug Console

Unesite 10 tekstova:
Unesite tekst 1: lev
Unesite tekst 2: nikolayevich tolstoy
Unesite tekst 3: fyodor mikhailovich
Unesite tekst 4: dostoevsky
Unesite tekst 5: dan
Unesite tekst 6: brown
Unesite tekst 7: isaac asimov
Unesite tekst 8: arthur c clark
Unesite tekst 9: jr tolkien john
Unesite tekst 10: green

Rezultat konkatencije i kapitalizacije prvih slova svake rijeci:
Lev Nikolayevich Tolstoy Fyodor Mikhailovich Dostoevsky Dan Brown Isaac Asimov Arthur C Clark Jr Tolkien John Green

C:\Users\kemal\source\repos\TestingGround\Debug\TestingGround.exe (process 9992) exited with code 0.
Press any key to close this window . . .
  
```

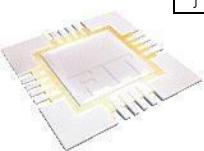
Zadatak 7

Neka je dat sljedeći prototip strukture datum:

```

struct Datum
{
    int dan;
    int mjesec;
    int godina;

    void unos();
    void ispis();
    bool isValid();
    int toInt();
};
  
```

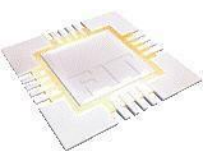


Uraditi implementaciju zadanih funkcija članica i po potrebi definirati i implementirati pomoćne globalne funkcije (tipa: `isLeap`, `getDaysForMonth` i sl.). Zatim, obavezno implementirati sljedeće globalne funkcije:

Funkcija	Ulazi	Izlaz	Definicija
<i>min</i>	<i>Datum & d1</i> <i>Datum & d2</i>	<i>Datum&</i>	Funkcija koja vraća stariji od dva datuma (po referenci).
<i>max</i>	<i>Datum & d1</i> <i>Datum & d2</i>	<i>Datum&</i>	Funkcija koja vraća noviji od dva datuma (po referenci).
<i>getDifferenceBetween</i>	<i>Datum & d1</i> <i>Datum & d2</i>	<i>int</i>	Funkcija koja računa razliku (u danima) između dva datuma. Voditi računa da <i>d1</i> ne mora biti nužno stariji datum od <i>d2</i> .
<i>getClosestDate</i>	<i>Datum * arr</i> <i>int size</i> <i>Datum & targetDatum</i>	<i>Datum *</i>	Funkcija koja za proslijeđeni datum <i>targetDatum</i> vraća njemu najbliži datum iz proslijeđenog dinamičkog niza objekata tipa datum. Povratna vrijednost je adresa objekta iz niza.

U main funkciji dodati sljedeće stavke:

- Unos veličine *vel* koja označava dimenzije dinamičkog niza objekata tipa *Datum*
- Kreiranje dinamičkog niza datuma
- Unos datuma (dana, mjeseca i godine) za svaki objekt u nizu (unos se ponavlja ako su datumi nevalidni)
- Ispis datuma
- Kreiranje objekta *targetDatum* i unos istog (unos se ponavlja ako je datum nevalidan)
- Pronalaženje najbližeg datuma iz unesenog niza pozivom funkcije *getClosestDate*
- Dealokacije dinamičkog niza



Microsoft Visual Studio Debug Console

```

Unesite dimenzije niza datuma:
5
-----
Unesite datum na lokaciji 0:
Unesite dan: 15
Unesite mjesec: 12
Unesite godinu: 2023
-----
Unesite datum na lokaciji 1:
Unesite dan: 17
Unesite mjesec: 12
Unesite godinu: 2023
-----
Unesite datum na lokaciji 2:
Unesite dan: 1
Unesite mjesec: 1
Unesite godinu: 2024
-----
Unesite datum na lokaciji 3:
Unesite dan: 2
Unesite mjesec: 2
Unesite godinu: 2024
-----
Unesite datum na lokaciji 4:
Unesite dan: 3
Unesite mjesec: 3
Unesite godinu: 2024
Datum 0: 15.12.2023
Datum 1: 17.12.2023
Datum 2: 1.1.2024
Datum 3: 2.2.2024
Datum 4: 3.3.2024
Unosenje novog datuma za pretragu...
Unesite dan: 31
Unesite mjesec: 12
Unesite godinu: 2023
Najblizi datum unesenom je:
1.1.2024

C:\Users\kemal\source\repos\TestingGround\Debug\TestingGround.exe (process 9768) exited with code 0.
Press any key to close this window . . .

```

Zadatak 8

Neka je dat sljedeći prototip strukture Glumac:

```

struct Glumac {
    char* ime = nullptr;
    char* prezime = nullptr;
    char drzava[40] = "";
    int brojFilmova = 0; // označava broj redova u dinamičkoj matrici
    int brojKorisnika = 0; // označava broj kolonu u dinamičkoj matrici
    int** ocjene = nullptr; // matrica dimenzija (brojFilmova × brojKorisnika)

    void unos(); // funkcija članica za alokaciju i unos vrijednosti
    float* getAveragesByUsers();
    void ispis();
    void dealokacije();
};

```



Uraditi implementaciju zadanih funkcija članica i po potrebi definirati i implementirati pomoćne globalne funkcije. Zatim, u main funkciji dodati sljedeće stavke:

- Kreiranje objekta tipa Glumac
- Poziv funkcije članice unos (funkcije koja radi alokaciju i konzolni unos)
- Poziv funkcije članice ispis
- Dohvaćanje niza prosječnih ocjena glumca za svakog korisnika (koristiti dinamičku memoriju)
- Poziv funkcije za dealokaciju objekta
- Dealokacija niza prosječnih ocjena.

Napomena: Nije dozvoljeno korištenje konstruktora i destruktora.

```

Microsoft Visual Studio Debug Console

Unesite ime:
Jim
Unesite prezime:
Carrey
Unesite drzavu:
Kanada
Unesite broj filmova:
3
Unesite broj korisnika:
5
Unesite ocjenu glumca za ulogu u filmu 0 od strane korisnika 0: 10
Unesite ocjenu glumca za ulogu u filmu 0 od strane korisnika 1: 10
Unesite ocjenu glumca za ulogu u filmu 0 od strane korisnika 2: 9
Unesite ocjenu glumca za ulogu u filmu 0 od strane korisnika 3: 9
Unesite ocjenu glumca za ulogu u filmu 0 od strane korisnika 4: 8

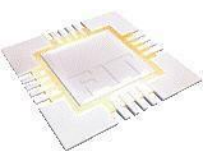
Unesite ocjenu glumca za ulogu u filmu 1 od strane korisnika 0: 8
Unesite ocjenu glumca za ulogu u filmu 1 od strane korisnika 1: 10
Unesite ocjenu glumca za ulogu u filmu 1 od strane korisnika 2: 10
Unesite ocjenu glumca za ulogu u filmu 1 od strane korisnika 3: 7
Unesite ocjenu glumca za ulogu u filmu 1 od strane korisnika 4: 9

Unesite ocjenu glumca za ulogu u filmu 2 od strane korisnika 0: 10
Unesite ocjenu glumca za ulogu u filmu 2 od strane korisnika 1: 9
Unesite ocjenu glumca za ulogu u filmu 2 od strane korisnika 2: 9
Unesite ocjenu glumca za ulogu u filmu 2 od strane korisnika 3: 9
Unesite ocjenu glumca za ulogu u filmu 2 od strane korisnika 4: 10

Ime i prezime: Jim Carrey
Drzava: Kanada
Broj filmova: 3
Broj korisnika: 5
-----
10 10 9 9 8
8 10 10 7 9
10 9 9 9 10
-----
Ispis prosjecnih ocjena glumca po svakom korisniku:
9.33333 9.66667 9.33333 8.33333 9





C:\Users\kemal\source\repos\TestingGround\Debug\TestingGround.exe (process 20248) exited with code 0.
Press any key to close this window . . .

```

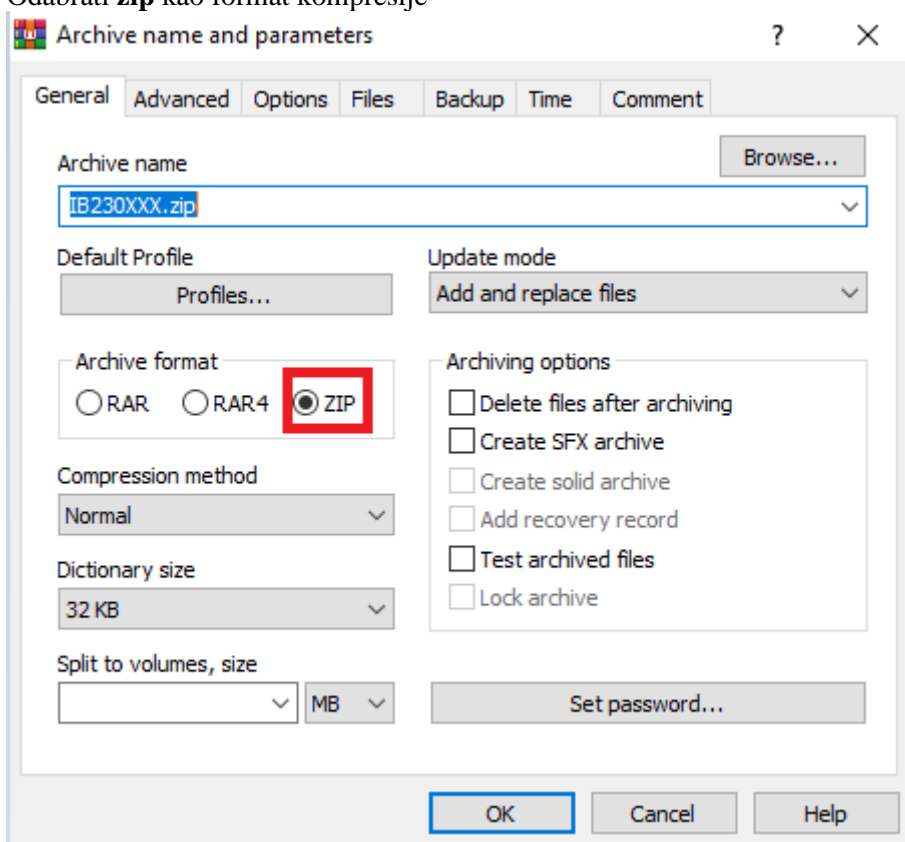


Kratke upute za spašavanje rješenja na ispitu [korištenje FTP servera]

1. Rješenja zadataka imenovati na sljedeći način:

Name	Status	Date modified	Type
 Zadatak 1.cpp		17. 11. 2023. 14:18	CPP File
 Zadatak 2.cpp		17. 11. 2023. 15:22	CPP File

2. Dodati sve *.cpp fajlove u jedan folder sa nazivom **IB230***** (broj indeksa studenta)
3. Uraditi kompresiju
 - a. Desni klik na folder i odabrati opciju *Add to archive*
 - b. Odabrati **zip** kao format kompresije



Rezultat kompresije je fajl sa nazivom **IB230***.zip**

4. Priključiti se na ftp server (<ftp.fit.ba>) preko File Explorer-a sa sljedećim login podacima:
 - Username: **student_km**
 - Password: **student_km**
5. Kopirati fajl **IB230***.zip** u odgovarajući folder (**Upload** sekcija)

Napomena: Da bi se određeni zadatak bodovao, potrebno je da se isti može uspješno pokrenuti i izvršavati. Ukoliko postoji greška prilikom kompajliranja, zadatak se označava kao 'NC' (Not Compiled), te se kao takav ne može bodovati. Isto tako, kardinalnom greškom se smatra nekontrolisano izvršavanje programa poput beskonačne petlje. U tom slučaju se zadatak, označava kao 'RE' (Runtime Error).

