

Име и презиме:
Индекс:
Паралелка:



ВТОР КОЛОКВИМ ПО ПРЕДМЕТОТ
Структури со програмирање
21.05.2014, Термин 1

1. Во кружна двојно поврзана листа се наоѓаат јазли чии инфо полиња се имиња на **играчи**. Во друга кружна двојно поврзана листа се наоѓаат јазли чии инфо полиња имаат вредност од **1 до 6** (во втората листа има повеќе јазли отколку во првата листа). Секој играч од првата листа фрла коцка и се движи низ втората листа за вредноста која паднала на коцката. Ако инфо бројот на јазолот на кој застанал се совпаѓа со вредноста која паднала на коцката, се бриши јазолот од втората листа, играчот останува во игра, а наредниот играч продолжува да брои од следниот јазол во втората листа. Ако инфо бројот на јазолот на кој застанал не се совпаѓа со бројот кој паднал на коцката, играчот испаѓа од игра. Играта продолжува сè додека не остане еден играч, кој се прогласува за победник.

Забелешка: случаен број од 1 до 6 се генерира со помош на кодот:

```
srand (time(NULL));
```

```
int sluchaen = 1 + (rand() % 6); //потребно е да се вклучат библиотеките stdlib.h и time.h
```

Пример :

Листа 1: |Дејан|<->|Тина|<->|Марија|<->|Јане|

Листа 2: |1|<->|6|<->|5|<->|3|<->|4|<->|1|<->|5|

Дејан фрла 4 Дејан останува во игра, |4| се брише од втората листа, се продолжува од |1|
Листа 1: |1|<->|6|<->|5|<->|3|<->|1|<->|5|

Тина фрла 2 Тина отпаѓа од игра, во втората листа се продолжува од |6|
Листа 2: |Дејан|<->|Марија|<->|Јане|

Марија фрла 6 Марија останува во игра, |6| се брише од втората листа, се продолжува од |5|
Листа 1: |1|<->|5|<->|3|<->|1|<->|5|

2. Да се напише класа **Prashanje** во која се чува редниот број и текстот за прашањето, како и податок за типот на прашање. Прашањето се внесува со помош на функцијата **vnesiPrashanje()** која како аргумент прима реден број, текст и тип на прашањето и креира прашање со внесените информации. Проверката на точноста на внесениот одговор од страна на корисникот се прави со помош на функцијата **proveriOdgovor()** во која како аргумент се проследува редниот број на точниот одговор за прашањата со повеќе понудени одговори, или точниот текст за дополнување за прашањата со дополнување.

На тестот постојат два типа на прашања: прашања со **повеќе понудени одговори** и прашања со **дополнување**. За прашањата со повеќе понудени одговори се чува бројот на понудени одговори, низа од понудените одговори (низа од стрингови), како и редниот број на точниот одговор во низата. За прашањата со дополнување се чува текстот на дадениот дел од одговорот, како и точниот текст за дополнување.

Ако типот на прашање е со повеќе понудени одговори, при внес на прашањето се проследува и информација за бројот на понудените одговори, понудените одговори и точниот одговор, додека ако прашањето е со дополнување, при внес се проследува текстот за дадениот дел од одговорот, како и текстот за дополнување.

Да се напише класа **Test** во која се чува низа од прашања и број на прашања.

Во класата Test да се напише функција **kreirajTest()**, која како аргумент прима број на прашања и ја полни низата од прашања.

Во класата Test да се напише и функција **proveriTest()**, во која од корисникот се бара за секое прашање (според типот) да внесе соодветен одговор и на крај ќе врати колку точни одговори внел корисникот.

```
int main()
{
    Test test;
    test.kreirajTest();
    int tochni = test.proveriTest();
    cout<<"Korisnikot pogodi vkupno "<<tochni<<" prashanja"<<endl; }
}
```

