

Egzamin z Programowania Komputerowego

4 lutego 2010

1. Napisać funkcję

```
bool anagram(char* s, char* t);
```

która zwraca true wtedy i tylko wtedy, gdy napis t można utworzyć z napisu s poprzez przestawienie liter. **(12 pkt.)**

2. Co zostanie wypisane na ekran w wyniku wykonania następującego programu: **(8 pkt.)**

```
void f(int n)
    int a=0;
    while(n) {
        if(n%10>a)
            a=n%10;
        n/=10;
    }
}
int main() { cout << f(100) << f(211) << f(873083) <<
f(666); }
```

Jaki wynik zwróci funkcja f dla dowolnego parametru?**(4 pkt.)**

3. Zaimplementować klasę `Las`, której obiekty reprezentują układy drzew. Każde drzewo ma unikalny numer (0..99) oraz współrzędne miejsca, w którym się znajduje. Zakładamy, że w lesie znajduje się (w danej chwili) nie więcej niż 100 drzew.**(12 pkt.)**

Metody publiczne:

- `Las()`

tworzy nowy, pusty las,

- `int sadz(double x, double y)`

dodaje drzewo o zadanych współrzędnych i zwraca jego unikalny, nadany przez funkcję numer. Jeśli w lesie jest już 100 drzew, to nic nie dodaje i zwraca -1,

- `void wytnij(int n)`

usuwa drzewo o podanym numerze jeśli takie istnieje

- `int ileDrzew()`

zwraca liczbę drzew w lesie

- `int najblizsze(int x, int y)`

zwraca numer drzewa, które jest położone najbliżej podanego punktu. Jeśli las jest pusty, zwraca -1.

4. Zaimplementować klasę `Scrabble`, której obiekty reprezentują zbiory liter (nie większe niż 7-elementowe; literki mogą się powtarzać).

Metody publiczne:

- `Scrabble()`

tworzy nowy, pusty zestaw,

- `bool dodaj(char l)`

jeśli w zestawie jest mniej niż 7 liter to dodaje literkę do zestawu i zwraca `true`. Jeśli nie, zwraca `false`,

- `bool uloz(char* s)`

jeśli z literek znajdujących się w zestawie można ułożyć napis `s`, to układa ten napis (tj. usuwa wszystkie literki, które są potrzebne do jego ułożenia). W przeciwnym przypadku zwraca `false`.

- `int liczbaZuzytychLiter()`

zwraca liczbę liter zużytych do układanie napisów. **(8 pkt.)**

Następnie zmodyfikować tą klasę tak, aby literki można było dodawać przy pomocy operatora `+=` i usuwać (układać) przy pomocy operatora `-=`. **(4 pkt.)**

5. Zaimplementować klasę `Tablica3`, której obiekty reprezentują tablice trójwymiarowe. **(12 pkt.)**

Metody publiczne:

- `Tablica3(int a, int b, int c)`

tworzy nową tablicę o zadanych wymiarach wypełnioną zerami

- `bool ustaw(int x, int y, int z, int w)`

wpisuje w miejscu o współrzędnych `x,y,z` wartość `w` i zwraca `true` o ile jest to możliwe, tzn. indeksy są poprawne. Jeśli indeksy nie mieszczą się w zakresach, to zwraca `false`.

- `int czytaj(int x, int y, int z)`

zwraca wartość znajdującą się w podanym miejscu; jeśli indeksy nie są poprawne to zwraca 0.

Uwaga: Jeśli to konieczne, napisać również destruktor.