

# Egzamin z Programowania Komputerowego

11 czerwca 2010

## 1. Napisać funkcję

```
void f(char* s);
```

która powoduje, że w napisie s wszystkie spacje zostają podwojone. Zakładamy, że w tablicy s jest dostatecznie dużo miejsca. **(12 pkt.)**

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

```
int f(int n)
    for(int k=n; k; k--)
        n=-n;
    return k+n;
}
int main() { cout << f(100) << f(211) << f(87) << f(666); }
```

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

## 3. Zaimplementować klasę Prosta, której obiekty reprezentują proste na płaszczyźnie. sss **(12 pkt.)**

### Metody publiczne:

- Prosta(Punkt a, Punkt b)

tworzy prostą przechodzącą przez punkty a i b,

- void przesun(double x, double y)

przesuwa prostą o zadany wektor

- bool zawiera(Punkt a)

zwraca true jeśli prosta zawiera dany punkt

- bool rowna(Prosta& p)

zwraca true jeśli parametr jest tą samą prostą

Następnie zmodyfikować tą klasę tak, aby porównanie prostych można było wykonywać przy pomocy operatora ==. **(4 pkt.)**

4. Zaimplementować klasę `Magazyn`, której obiekty reprezentują magazyn. W magazynie znajdują się towary – każdy z nich ma nazwę i ilość (liczbę całkowitą). Można założyć, że w magazynie nie pojawi się więcej niż 100 różnych towarów.

**Metody publiczne:**

- `Magazyn()`

tworzy nowy, pusty magazyn,

- `void dodaj(char* towar, int ilosc)`

dodaje pewną ilość towaru,

- `int usun(char* towar, int ilosc)`

usuwa pewną ilość towaru o podanej nazwie. Jeśli ilość danego towaru jest mniejsza, usuwa wszystko. Zwraca ilość pozostałego towaru.

- `int ile(char* towar)`

zwraca ilość towaru o zadanej nazwie. **(12 pkt.)**

5. Zaimplementować klasę `Wektor`, której obiekty reprezentują tablice liczb. **(12 pkt.)**

**Metody publiczne:**

- `Wektor(int a)`

tworzy nową tablicę o zadanej długości wypełnioną zerami.

- `bool ustaw(int i, int w)`

wpisuje w miejscu o indeksie `i` wartość `w` i zwraca `true` o ile jest to możliwe, tzn. indeks jest poprawny. W przeciwnym wypadku zwraca `false`.

- `int czytaj(int x)`

zwraca wartość znajdującą się w podanym miejscu; jeśli indeks nie jest poprawny to zwraca 0.

- `void ustawRozmiar(int a)`

ustawia nowy rozmiar tablicy. Jeśli rozmiar się zwiększa, nowe pola powinny zostać wypełnione zerami. Wartości z tablicy powinny zostać zachowane (o ile to możliwe)