

Przykładowe pytania na kolokwium z Języków programowania

Sprawdź czy znasz i rozumiesz ważne pojęcia:

- programowanie
- kompilacja i konsolidacja

- typ i literał, typ wbudowany i typ użytkownika
- obiekt, zmienna, stała
- deklaracja i inicja(liza)cja zmiennej
- bezpieczeństwo typów

- technika „dziel i zwyciężaj”, technika abstrakcji
- wyrażenie, operator i operand
- instrukcje: warunkowe (selekcja), pętle (co to jest iteracja?)
- funkcja: wywołanie funkcji, zwracanie wartości, przepływ sterowania
- kontener danych

- błędy: kompilacji, konsolidacji, czasu wykonania, logiczne
- wyjątki
- odrobaczanie (debugging)
- warunki początkowe i końcowe, przypadki brzegowe

- analiza, projekt, implementacja, testowanie
- pseudokod

- pliki nagłówkowe
- zakres (globalny, klasy, lokalny, instrukcji)
- deklaracja i definicja funkcji
- przekazywanie parametrów funkcji (przez wartość, referencję i stałą referencję)
- stała magiczna

- typy użytkownika: klasa i struktura
- konstruktor i inne składowe klasy (struktury)

- wejście i wyjście programu
- strumień wejścia i wyjścia
- plik (tekstowy, binarny)

Przykładowe zadania

1. Uzupełnij podpisy, połącz typy i odpowiadające im przykłady literałów, określ czy typ jest wbudowany:

wbudowany?
.....	complex<double>	false
.....	bool	'a', '\n', '('
.....	char	complex<double>(12.3,99), complex<float>(1.3)

2. Wyjaśnij na czym polega technika „dziel i zwyciężaj”

.....

.....

.....

3. Na czym polegają błędy czasu wykonania (*run-time errors*)

.....
.....
.....

4. Wymień etapy tworzenia oprogramowania, opisz krótko jeden z nich

-, tzn.:
.....
.....
-
-

5. Uzasadnij celowość tworzenia typów użytkownika (klas i struktur):

.....
.....
.....

6. Określ słowami warunki początkowe i końcowe następującego kodu:

```
int pole(int wysokosc, int szerokosc) {  
    // Warunki początkowe, czyli czego funkcja wymaga aby wykonać poprawne obliczenia:  
    .....  
    .....  
    int p = wysokosc*szerokosc;  
    // Warunki końcowe, czyli co funkcja obiecuje jeżeli obliczenia wykonane poprawnie:  
    .....  
    .....  
    return p;  
}
```

7. Opisz poniższy kod w pseudokodzie

```
vector<int> bubble_sort(const vector<int>& w) .....  
{ .....  
    if (w.size()<2) return w; .....  
    .....  
    vector<int> v = w; .....  
    .....  
    int n = v.size(); .....  
    while (n>1) { .....  
        for (int i=0;i<n-1;++i) .....  
            if (v[i]>v[i+1]) { .....  
                int tmp = v[i]; .....  
                v[i] = v[i+1]; .....  
                v[i+1] = tmp; .....  
            } .....  
        --n; .....  
    } .....  
    return v; .....  
}
```

8. Na osiach życia zmiennych i stałych zaznacz, w których zakresach są dostępne (pogrubienie).

Ponadto zaznacz znakiem „-” utworzenie, a znakiem „X” usunięcie zmiennej/stalej z pamięci.

```

const int n=10;.....
.....
int largest(vector<int> v).....
    int r = 0; .....
    for (int i = 0; i<v.size(); ++i) .....
        r = max(r,abs(v[i])); .....
    return r;.....
} .....
.....
int main() { .....
    vector<int> w; .....
    int tmp; .....
    for (int i=0; i<n; ++i) { .....
        cin >> tmp;.....
        w.push_back(tmp); .....
    } .....
    cout << "Bezwzględnie największy element: " .....
        << largest(w) << endl; .....
} .....

```

9. Nazwij elementy składowe klasy Książka i uzupełnij komentarz

```

class Książka {
    string k_autor; // to sq ..... składowe
    string k_tytul;
    int k_rok;
public:
    Książka(const string& autor, const string& tytul, int rok); // ta specjalna funkcja to .....
    void wyswietl() // funkcja składowa
        { cout << k_autor << ". " << k_tytul << " (" << k_rok << ")"; }
};

Książka::Książka(const string& autor, const string& tytul, int rok)
    : k_autor(autor), k_tytul(tytul), k_rok(rok)
{
    // Sprawdzamy warunki ..... :
    if (autor.empty() || tytul.empty() || rok<1455)
        throw runtime_error("Niewlasciwe argumenty!");
}

```

10. Uzupełnij tabelę, wpisując wartości zmiennych oraz to co jest wypisywane na ekran: przed wejściem do pętli, przy każdej iteracji pętli i po wyjściu z pętli. Jeśli zmienna jest niezadeklarowana w danym zakresie - wpisz „-”

	Przed pętlą	Kolejne iteracje pętli			
i					
ile_ujemnych					
ekran					

```

vector<int>v=[2,-3,1,0];
int ile_ujemnych=0;
for (int i=0; i<v.size(); i++) {
    cout<<v[i]<<endl;
    if (v[i]<0) ile_ujemnych+=1;
}
cout<<ile_ujemnych;

```

Proszę się też spodziewać zadań typu: Znajdź i opisz błędy w kodzie

Podział na grupy 30/11

Osoby o nazwiskach zaczynających się na litery od A do N, godz. 13.15

Osoby o nazwiskach zaczynających się na litery od O do Ż, godz. 14.15

Uwagi

- Przewiduję 6 zadań na kolokwium
- Obowiązuje zasada „zero tolerancji” dla ściągania (bez ostrzeżeń i bez możliwości odwołania)
- Na kolokwium proszę odpowiadać tylko na pytania, na które rzeczywiście znają Państwo odpowiedź:
 - w zadaniach polegających na uzupełnieniu dostaną Państwo zero punktów, gdy liczba nieprawidłowych odpowiedzi przekroczy liczbę prawidłowych!
 - w zadaniach polegających na wyjaśnieniu punkty zostaną przyznane tylko, gdy odpowiedź będzie jasna i klarowna.