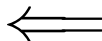


Písomná skúška z PT: 25. januára 2017

Vypni zvonenie na mobilnom telefóne!

Táto písomná skúška trvá 90 minút. Počas skúšky je povolené používať knihy, poznámky, učebné texty, referencie jazyka C/C++. Fotenie tohto dokumentu počas skúšky je zakázané.

Meno a priezvisko:



1. Operácia `find` v *nevyváženom* binárnom strome má v najhoršom prípade zložitosť (uved' najpresnejšie ohraničenie): (4 body)

Riešenie:

- A $O(\log n)$
- B $O(n)$
- C $O(1)$
- D $O(n^{\frac{1}{2}})$

2. Nech n je počet prvkov, ktoré musia byť roztriedené. Časová zložitosť merge sortu je (uved' najpresnejšie ohraničenie): (4 body).

- A $O(1)$
- B $O(\log n)$
- C $O(n \log n)$
- D $O(n^2)$

3. Dijkstrov algoritmus pre najkratšiu cestu je možné použiť len v prípade, ak má graf s n uzlami: (4 body)

- A záporne a kladne ohodnotené hrany
- B hĺbku max. $O(n)$
- C hĺbku max. $O(\log n)$
- D kladne ohodnotené hrany

4. Dijkstrov algoritmus pre najkratšiu cestu vyžaduje operáciu: (8 bodov)

- A Extract maximum
- B Decrease key
- C Increase key
- D Balance

Písomné odôvodnenie riešenia (min. 10 slov, max. 20 slov):

5. Nech m je `std::map<int, double>` a n je počet prvkov v m , potom `m[3] = 3.14`; má v najhoršom prípade zložitosť (uved' najpresnejšie ohraničenie): (4 body)

- A $O(1)$
- B $O(\log n)$
- C $O(n \log n)$
- D $O(n^2)$

6. Nech m je `std::map<int, double>` a n je počet prvkov v m , potom `double d = m.at(3)`; má v najhoršom prípade zložitosť (uved' najpresnejšie ohraničenie): (4 body)

- A $O(1)$
- B $O(\log n)$
- C $O(n \log n)$
- D $O(n^2)$

7. Nech s je `std::set<long double>` a n je počet prvkov v s , potom `std::set<long double>::iterator it = s.find(3.14L)`; má v najhoršom prípade zložitosť (uved' najpresnejšie ohraničenie): (4 body)

- A $O(1)$
- B $O(\log n)$
- C $O(n \log n)$
- D $O(n^2)$

8. Čo sa vypíše na obrazovku? (4 body)

```
#include<iostream>
#include<string>
#include<vector>
using namespace std;

int main(){
    vector<string> v;
    v.push_back("Imro");
    v.push_back("Fero");
    v.push_back("");
    cout << v[3] << endl;

    return 0;
}
```

Riešenie:

- A Imro a new_line
- B Fero a new_line
- C nedá sa skompilovať
- D nastane pamäťová chyba

Písomné odôvodnenie riešenia (min. 10 slov, max. 20 slov):

9. Čo sa vypíše na obrazovku? (4 body)

```
#include<iostream>
#include<string>
#include<vector>
using namespace std;

int main(){
    vector<string> v;
    v.push_back("Imro");
    v.push_back("Fero");
    v.push_back("");

    try{
        cout << v[3] << endl;
    } catch (...) {
        cout << v[2] << endl;
    }

    return 0;
}
```

Riešenie:

- A new_line
- B Fero a new_line
- C nedá sa skompilovať
- D nastane pamäťová chyba

Písomné odôvodnenie riešenia (min. 10 slov, max. 20 slov):

10. Čo sa vypíše na obrazovku? (8 bodov)

```
#include<iostream>
using namespace std;

class Google {
```

```

private:
    double d;

public:
    double* get(){
        double d;
        return &d;
    }
    void set(double d){
        this->d = d;
    }
};

int main(){
    Google* g = new Google();

    g->set(3.14);
    double* d = g->get();

    cout << *d << endl;

    delete g;
    return 0;
}

```

Riešenie:

- A 3.14000 a new_line
- B 3.14 a new_line
- C nastane pamäťová chyba
- D nedá sa skompilovať

Písomné odôvodnenie riešenia (min. 10 slov, max. 20 slov):

11. Čo sa vypíše na obrazovku? (8 bodov)

```

#include<iostream>
using namespace std;

class Google {

private:
    double d;

public:
    double* get(){
        double d;
        return &d;
    }
    void set(double d = 3.14){
        this->d = d;
    }
};

class Yahoo : public Google {

private:
    double d;

public:
    double get(){
        return d;
    }
};

```

```

    }
    void set(double d = -3.14){
        this->d = d;
    }
};

```

```

int main(){
    Yahoo* y = new Yahoo();

    cout << y->get();

    delete y;
    return 0;
}

```

Riešenie:

- A 3.14000
- B 0
- C -3.14000
- D nastane pamäťová chyba

Písomné odôvodnenie riešenia (min. 10 slov, max. 20 slov):

12. Čo sa vypíše na obrazovku? (8 bodov)

```

#include<iostream>
#include<string>
using namespace std;

class Auto {

private:
    string meno;

public:
    Auto(string meno="Porsche"){
        this->meno = meno;
    }

    string getMeno(){
        return this->meno;
    }

    virtual string operator!() {
        return (this->meno).append(".");
    }
};

class Automobil : public Auto {

public:
    Automobil(string meno = "Lada"){

        string operator!() {
            return (this->getMeno()).append("!");
        }
    }
};

int main(){
    Automobil avto;

    cout << !avto; cout << !avto << endl;
}

```

```
    return 0;
}
```

Riešenie:

- A Lada.Lada.
- B Lada!Lada!
- C Porsche.Porsche.
- D Porsche!Porsche!

Písomné odôvodnenie riešenia (min. 10 slov, max. 20 slov):

13. Čo sa vypíše na obrazovku (kompilované s g++ -std=c++11)? (8 bodov)

```
#include <memory>
#include <iostream>
#include <vector>

using namespace std;

class PT{
private:
    int students;

public:
    PT(){
        students=0;
    }
};

int main(){
    vector<shared_ptr<PT> > vec;

    for(int i=0; i < 2 ; i++){
        shared_ptr<PT> myPtr= make_shared<PT>();
        vec.push_back(myPtr);
    }

    for(auto item: vec){
        cout<<item.use_count()<<" ";
    }

    return 0;
}
```

Riešenie:

- A 0 0
- B 1 1
- C 2 2
- D nastane pamäťová chyba

Písomné odôvodnenie riešenia (min. 10 slov, max. 20 slov):

14. Napíš pseudokód pre výpočet najdlhšej najkratšej cesty, ktorá neobsahuje uzol s väčším stupňom ako k . Funkcia `shortest_path(node)` vráti najkratšiu cestu z uzla `node` ku všetkým ostatným uzlom; túto funkciu nie je treba definovať. Pseudokód napíš na druhú stranu! Max. 10 riadkov! (8 bodov)