Вариант Э2\_2015 Ф.И.О.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_№ группы\_\_\_\_\_\_\_\_

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|  |  |  |  |  |  |  |  |  |  |

***NB!* Во всех задачах, где это требуется, предполагается наличие необходимых включений и директивы using namespace std.**

1. При компиляции функции main() следующей программы будет зафиксирована ошибка: "неразрешимый вызов функции f ('a', 0)". Объяснить причину такой ошибки, пользуясь алгоритмом выбора best-matching функции.

**struct** A { **int** main () {

**operator int** () { **return** 1; } A a ;

}; f ('a', 0);

**void** f (**double** d, **char** c) { cout << "f (double, char)\n";} **return** 0;

**void** f (**double** d, **int** j) { cout << "f (double, int)\n";} }

**void** f (A a, **const char \*** p) { cout << "f( A, const char \*)\n";}

**void** f **(int** i, **const char \*** p) { cout << "f (int, const char \*)\n";}

**2.** Привести пример программы, в которой в результате вызова из функции *main()* метода *f()* некоторого класса через указатель *p* ( ***p -> f()*** ) проработает метод ***f()***, описанный в **private** – области другого класса.

**3.** При компиляции приведенной программы на С++ фиксируются ошибки. Объяснить, в чем они заключаются. Исправить все ошибочные конструкции **только** путем **вставки** или **замены** служебных слов С++. Что будет напечатано в результате работы получившейся правильной программы?

|  |  |
| --- | --- |
| **struct** A {  **virtual** **void** h (**int** & a) {  a++;  cout << "h\_from\_A\n"; }  **virtual int** t () {  cout << "t\_from\_A\n "; **return** 0;}  };  **struct** S : A {  **int** h (long & a) {  a++;  cout << "h\_from\_S\n"; **return** 0; }  **int** t () {  cout << "t\_from\_S\n "; **return** 0; }  }; | **int** main() {  **int** n = 5;  **const** A a1;  S s1;  A \* p = &s1;  a1.t ();  p -> h (n);  p -> t ();  cout << "n = " << n << endl;  **return** 0;  } |

**4.** Что будет напечатано в результате выполнения программы? Компилятор оптимизирующий.

|  |  |
| --- | --- |
| **struct B {**  B(){ cout << "B()\n"; }  ~B(){ cout << "~B()\n"; }  **};**  **class** A {  **static** B sa;  **public**:  A(){ cout << "A()\n"; }  A(**const** A & a){ cout << "A(const A&)\n"; }  ~A(){ cout << "~A()\n"; }  }; | B A::sa;  **int** main() {  **try** {A a1;  **try** { **throw** B(); }  **catch** (A &) { cout << 333 << endl; }  }  **catch** (...) { cout << " ... \n"; }  cout << " End\n ";  **return** 0;  } |

**5.** Написать **алгоритм в стиле STL**, который применим к контейнерам, имеющим двунаправленый итератор (BidirectionalIterator). Алгоритм должен распечатывать 3-й с конца элемент заданного диапазона контейнера. Если количество заданных элементов меньше 3, напечатать первый заданный элемент или строку "Empty\n" в случае пустого диапазона.

**6.** **a)** Построить приведенную грамматику, эквивалентную заданной грамматике G:

*S* → *aAb* | *CB*

*B* → *bB*

*A* → *aA* | *D*

*С* → *сСс* | *B* | *c*

*D* → *d*

**б)** Каков тип получившейся грамматики?

**в)** Какой язык порождает грамматика (формула)?

**г)** Каков тип этого языка?

*Для типов грамматики и языка указать максимально возможный номер по Хомскому*.

**7.** Дана автоматная леволинейная грамматика G:

*S* → *C*⊥

*A* → *Aa* | *a*

*B* → *Aa* | *b*

*C* → *Ba*

**а)** Детерминирован ли разбор по этой грамматике? **Обосновать ответ**.

**б)** Построить по грамматике конечный автомат и преобразовать его к детерминированному

виду по алгоритму НКА → ДКА

**в)** Построить по получившемуся ДКА праволинейную автоматную грамматику.

**г)** Эквивалентны ли исходная и получившаяся автоматные грамматики? **Обосновать ответ**.

**8.** Дана КС-грамматика G:

*S* → *A* | *B*

*A* → *aA* | ε

*B* → *bB* | *cC* | ε

*C* → *aABC | c*

**а)** Преобразовать грамматику G к неукорачивающей КС-грамматике с помощью алгоритма

устранения пустых правых частей в КС грамматике (КС → НКС).

**б)** Применим ли метод рекурсивного спуска к исходной грамматике? **Ответ обосновать**.

**9.** Построить грамматику выражений, содержащих

- левоассоциативные целочисленные бинарные операции **+** и **–** ;

- односимвольные целые числа **0, 2, 4, 6** ;

- скобок нет.

Добавить в грамматику действия (только вида **cout <<"символ";** ) по переводу заданных выражений в ПОЛИЗ во время анализа РС-методом.

Приоритет операций **+** и **–** должен быть **задан построенной грамматикой** таким образом, чтобы ПОЛИЗ выражения:

**2 + 4 – 6 – 0 + 4 + 2**

совпадал с ПОЛИЗом выражения со скобками:

**((2+((4–6)–0)+4)+2)**.

1. **а)** Назвать основные задачи редактора связей.

**б)** Привести пример ситуации, в которой редактор связей выдаст сообщение об ошибке.