Вариант Э1\_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 два примера классов, для которых нельзя создать объект. Причины, по которым нельзя создать объект, должны быть для этих классов **разными.**

**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** B { **virtual** **char** f () { cout << "f\_from\_B\n"; **return** 0; } **virtual void** g () { cout << "g\_from\_B\n "; }};**struct** Т : B { **int** f () { cout << "f\_from\_Т\n"; **return** 0; }  **void** g () { cout << "g\_from\_Т\n "; }}; |  **int** main () { **int** n; **const** Т ct; T t; B \* pb = &t; n = pb -> f (); pb -> g (); сt.g(); **return** 0; } |

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

|  |  |
| --- | --- |
| **class** A { **public**: **static** A sa; A(){ cout << "A()\n"; } A(**const** A & a){ cout << "A(const A&)\n"; } ~A(){ cout << "~A()\n"; }}; | A A::sa; **int** main() {  **try** { A a1 = A::sa; **try** { **throw** A(); }  **catch**(**int**){ cout << 111 << endl; } } **catch** (...) { cout << " ... \n"; } cout << 222 << endl; **return** 0; } |

**5.** Написать на С++ **алгоритм в стиле STL**, который применим к контейнерам, имеющим итератор произвольного доступа (Random\_accessIterator). Алгоритм должен распечатывать элементы с 10-го по 5-й из заданного диапазона контейнера (элементы внутри диапазона нумеруются с единицы). Если количество заданных элементов меньше 10, напечатать последний элемент диапазона или 0 в случае пустого диапазона.

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

*S* → *aABb* | *C*

 *B* → *bB* | ε

 *A* → *aA* | *D*

 *С* → *сСс* | *B*

 *D* → *dD*

 **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 праволинейная грамматика G:

*S* → 1*B*

*B* → 1*B* | 1*A*

 *A* → 1*C*

 *C* → 0*A* | ⊥

**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 леволинейную автоматную грамматику.

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

**8.** Дана КС-грамматика G:

*S* → *aABc* | *C*

*A* → *aB* | *bA*

 *B* → *bBC* | ε

 *C* → *cC* | ε

**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, 1, 2, 3, 4, 5**;

- 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:

 **5 \* 3 / 2 / 1 \* 2 \* 1**

 совпадал с ПОЛИЗом выражения со скобками:

**( ( 5 \* ( ( 3 / 2 ) / 1 ) \* 2 ) \* 1 )**.

1. a) Перечислить этапы **создания** программного продукта.

б) Что называют программным продуктом?