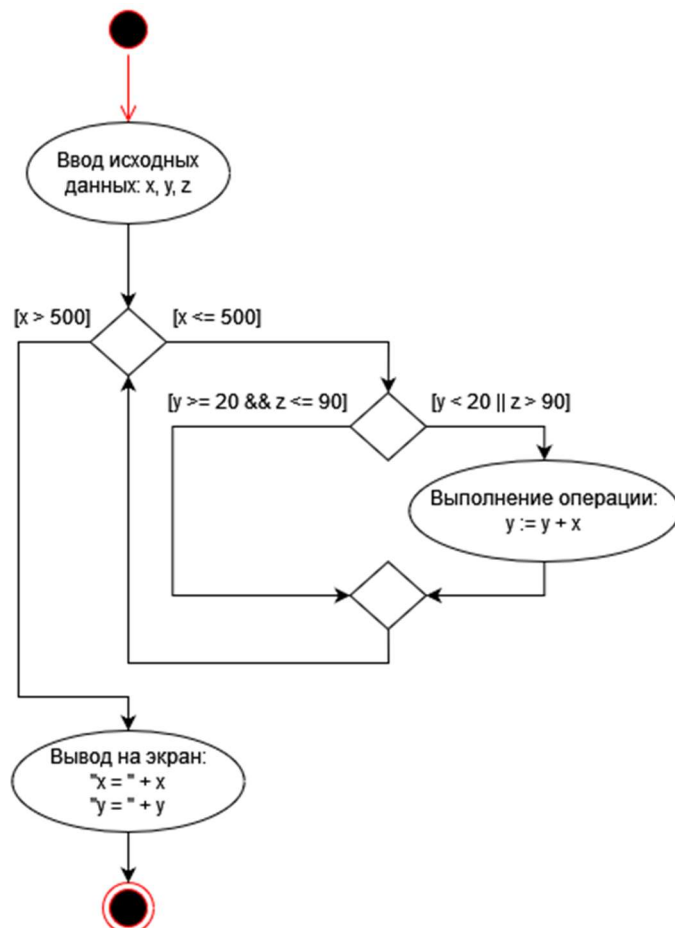


**Контрольная работа №1**  
**Тема «Основы построения диаграмм UML»**

**Вариант №13**

**Задача №1.** Построить алгоритм работы следующей программы на языке Java в виде диаграммы деятельности UML, описывающей ввод требуемых данных, необходимые вычисления и вывод информации на экран:

```
int x, y, z;  
Scanner sc=new Scanner(System.in);  
System.out.print("x=");  
x=sc.nextInt();  
System.out.print("y=");  
y=sc.nextInt();  
System.out.print("z=");  
z=sc.nextInt();  
while (x<=500) {  
    if (y<20 || z>90)  
        y+=x;  
    x+=15;  
}  
System.out.println("x="+x);  
System.out.println("y="+y);
```



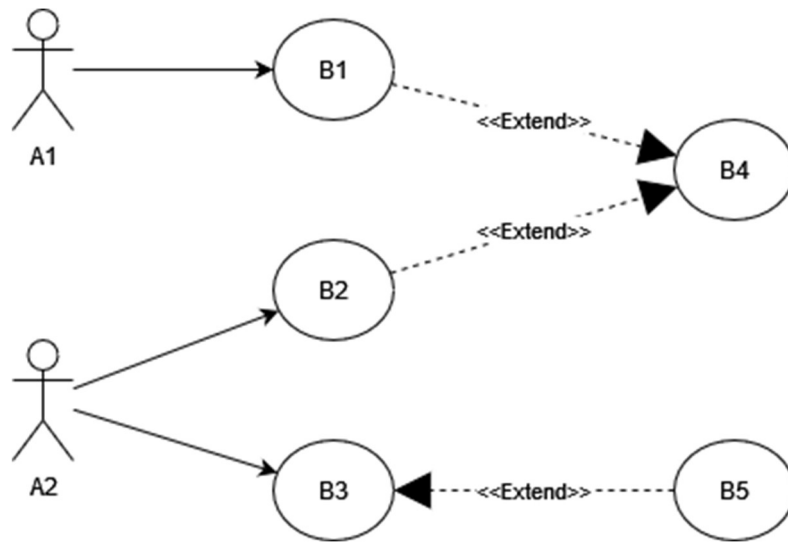
**Задача №2.** Нарисовать диаграмму вариантов использования UML по данному описанию:

Актер *A1* инициирует выполнение варианта использования *B1*.

Актер *A2* инициирует выполнение вариантов использования *B2* и *B3*.

Варианты использования *B1* и *B2* являются потомками варианта использования *B4*.

Вариант использования *B5* является расширяющим по отношению к варианту использования *B3*.



Пояснения:

1. В теоретических материалах слово «инициализирует» было написано лишь единожды, как синоним направленной ассоциации, поэтому такая используется в данной диаграмме.
2. В теории ООП «расширять» и «наследоваться от» - понятия идентичные. Например, в языке Java подобная строка говорит о наследовании:

```
class Child extends Parent { /*Block*/ }
```

3. Если *B5* является расширяющим, то он наследник, а расширяемый *B3* – родителем.