Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра конструирования и технологии радиоэлектронных средств
Кафедра теоретических основ радиотехники

Паспорт зачета

по дисциплине «Информационные технологии и основы программирования», 2 семестр

1. Методика оценки

Зачет проводится в письменной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-9, второй вопрос из диапазона вопросов 10-22, третий вопрос из диапазона вопросов 23-36, четвертый вопрос из диапазона вопросов 37-57 (список вопросов приведен ниже).

Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет РЭФ

Билет № 1
к зачету по дисциплине «Информационные технологии и основы программирования»

1. Дать определение. Инкапсуляция – это …?
2. Конструктор класса. Привести пример создания конструктора класса.
3. Инструкции управления break, continue, return
4. Привести результат вычислений double x=10.0; y=3.0; int i =(int)x/y; i=?

Утверждаю: зав. кафедрой ____ _____ должность, ФИО
(подпись)
(дата)

2. Критерии оценки

- Ответ на билет для зачета считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, при решении задачи допускает принципиальные ошибки, оценка составляет *0-9 баллов.*
- Ответ на билет для зачета засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, при решении задачи допускает непринципиальные ошибки, например, вычислительные, оценка составляет *10-12 баллов.*
- Ответ на билет для зачета билет засчитывается на **базовом** уровне, если студент при ответе на вопросы формулирует основные понятия, проводит анализ причин, условий,

не допускает ошибок при решении задачи, оценка составляет *13-17 баллов.*

- Ответ на билет для зачета билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, не допускает ошибок и способен обосновать выбор метода решения задачи, оценка составляет 18-20 *баллов.*

3. Шкала оценки

Зачет считается сданным, если сумма баллов по всем заданиям билета оставляет не менее 10 баллов (из 20 возможных).

В общей оценке по дисциплине баллы за зачет учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. **Вопросы к** зачету **по дисциплине «**Информационные технологии и основы программирования**»**

Объектно-ориентированное программирование

1. Дать определение. Класс – это …?
2. Привести пример создания класса в тексте программы
3. Дать определение. Инкапсуляция – это …?
4. Привести пример инкапсуляции в тексте программы
5. Дать определение. Полиморфизм – это …?
6. Привести пример полиморфизма в тексте программы
7. Дать определение. Наследование – это …?
8. Привести пример наследования в тексте программы
9. Привести пример создания открытых и закрытых членов класса. Пояснить понятие закрытых и открытых членов класса.

Структура языка программирования

10. Перегрузка функций. Привести пример создания перегрузки функций. Для чего служит перегрузка функций.
11. Конструктор класса. Привести пример создания конструктора класса.
12. Деструктор класса. Привести пример создания деструктора класса.
13. Что означает ссылочный тип данных. Привести пример создания ссылочного типа данных
14. Что означает типы значений (простые типы). Привести пример создания типа значений
15. Целочисленные типы данных
16. Типы значений для представлений чисел с плавающей точкой
17. Символьные типы данных
18. Логические типы данных
19. Инициализация переменных. Привести пример инициализации переменных.
20. Области видимости данных
21. Преобразование и приведение типов
22. Тип члена класса «Свойство». Создание свойств. Привести пример создания класса со свойством.

Инструкции языка программирования C#

23. Инструкции управления ветвлением в языке C#
24. Инструкции управления циклом в языке C#
25. Инструкции управления break, continue, return
26. Инструкция управления “?”
27. Какой процесс называется рекурсивным?
28. Привести примеры создания одномерного массива его инициализации.
29. Привести примеры создания двумерного массива его инициализации.

30. Привести примеры создания многомерного массива его инициализации.
31. Привести примеры создания «рваного» массива его инициализации.

Операторы языка C#

32. Назовите арифметические операторы языка программирования C#
33. Назовите операторы отношений языка программирования C#
34. Назовите логические операторы языка программирования C#
35. Назовите операторы присваивания языка программирования C#
36. Назовите поразрядные операторы языка программирования C#

Задачи

37. double x=10.0; y=3.0; int i =(int)x/y; i=?
38. int i =255; byte b=(byte) i; b=?
39. int i =257; byte b=(byte) i; b=?
40. uint u=3200; short s=(short)u; s=?
41. uint u=6400; short s=(short)u; s=?
42. long l=6400; uint u=(uint)l; u=?
43. char ch1 = ‘1’; char ch2=’2’; char ch3 = ch1+ch2; ch3=?
44. int x=10; int y=x++; x=? y=?
45. int x=10; int y=++x; x=? Y=?;
46. int x=5; int y=6; int z=x+++++y; z=? x=? y=?
47. int x=3; int y=4; int z=++x+y; z=? x=? y=?
48. int x=5; int y=6; int z=x+++y; z=? x=? y=?

49. void function()
    {
    int x=10;
    {
    int x=10;
    x++;
    }
    int y=x;
    }
    y=?

50. int x=10; y=1;
    switch(x)
    {
    case 10 : y++; break;
    case 20 : y=y+2; break;
    }
    x=?, y=?
    7.12.
    int x=10; y=1;
    switch(x)
    {
    case 10 : y++;
    case 20 : y = y + 2;
    default : y = y + 3;
    }
    x=?, y=?
51. int x=20; y=1;
    switch(x)
    {

```
case 10 : y++;
case 20 : y=y + 2;
case 30 : y=y + 3;
case 40 : y=y + 4;
}
x=?, y=?
```

52.
```
int y=0;
for(int i=0; (i<10)||( y<3); i++)
{
i++;
y++;
}
i=?, y=?
```

53.
```
for(int i=0, y=0; i<10; ++i)
{
if((i%2) == 0)
continue;
y++;
}
i=?, y=?
```
7.17.
```
for(int i=0, y=0; i<10; ++i)
{
if((i%2) == 0)
break;
y++;
}
i=?, y=?
```

54.
```
for(int i=0, y=0; i<10; ++i)
{
if((i%2) == 0)
return;
y++;
}
i=?, y=?
```

55.
```
int x=1, y=1;
while(x<10)
{
x++;
if((x%y) != 0)
y++;
}
x=?, y=?
```

56.
```
int x=1, y=1;
```

```
do
{
if((x%y)==0)
y++;
}while(x<10);
int function(int x, int y)
{
if(x>100) return 4;
return function(y, x+y);
}
y=function(1,1);
```

Какие значения будет возвращать функция function(…)? y=?

57.
```
int function(int x)
{
if(x>100) return 5;
return function(x*x);
}
y=function(1);
```
y=?