Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра электроники и электротехники

Паспорт зачета

по дисциплине «Информационные технологии в профессиональной деятельности»,
3 семестр

1. Методика оценки

Зачет проводится в письменной форме, по билетам. Билет состоит из 5 вопросов и задачи и формируется по следующему правилу:

первое задание выбирается из диапазона, второе задание - из диапазона вопросов 5 – 8, третий вопрос – 9-11, четвертое задание из диапазона 12-16, 5 задание из диапазона 17-21, задача из диапазона 22-27.

- первый вопрос выбирается из диапазона вопросов 1 - 4;
- второй вопрос - 5 – 8;
- третий вопрос - 9 – 11;
- четвертый – 12-16;
- пятый – 17-22;
- задача – 23-28.

Таким образом, проверяются результаты обучения по дисциплине, соотнесенные с установленными в программе индикаторами достижения компетенций.

На зачете преподаватель вправе задавать студенту уточняющие и дополнительные вопросы из общего перечня (п. 4).

Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет РЭФ

Билет № _____
к зачету по дисциплине «Информационные технологии в профессиональной деятельности»

---

1. Вопрос 1
2. Вопрос 2
3. Вопрос 3
4. Вопрос 4
5. Вопрос 5
6. Задача

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

2. Критерии оценки результатов обучения, соотнесенных с уровнями освоения индикаторов достижения компетенций

- Ответ на билет для зачета билет засчитывается на **продвинутом** уровне, если студент при ответе на задания дает верные ответы, предлагает оптимальное и верное решение задачи, оценка составляет 18-20 *баллов.*
- Ответ на билет для зачета билет засчитывается на **базовом** уровне, если дает верные ответы на задания, не допускает ошибок при решении задачи, оценка составляет *14-17 баллов.*
- Ответ на билет для зачета засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает точные ответы на большинство заданий, знает, как работают управляющие конструкции, умеет создавать функции, при решении задачи допускает непринципиальные ошибки, например, негрубые синтаксические, или решает задачи частично, оценка составляет 10-13 *баллов.*
- Ответ на билет для зачета считается **неудовлетворительным**, если студент при ответе на вопросы не дает точных ответов, не способен понимать текст программ, не знает принцип работы основных управляющих конструкций, при решении задачи допускает принципиальные ошибки или не решает задачу, оценка составляет *менее 10 баллов.*

3. Шкала оценки

В таблице 1 приведена шкала оценки каждого задания билета

Таблица 1

№ задания	1	2	3	4	5	6
Баллы	2	3	4	3	3	5

Зачет считается сданным, если сумма баллов по всем заданиям билета составляет от 10 до 20 баллов включительно. Сумма менее 10 баллов признается неудовлетворительным результатом промежуточной аттестации по дисциплине.

В общей оценке по дисциплине баллы за зачет учитываются в соответствии с правилами балльно-рейтинговой системы, установленными в НГТУ.

4. Вопросы к зачету по дисциплине «Информационные технологии в профессиональной деятельности»

1. В программе описан одномерный целочисленный массив с индексами от 0 до n. Ниже представлен фрагмент программы, обрабатывающей данный массив:

```
int s = n;
int z = A[0];
for ( i = 1 ; i<= n ; i ++)
if(A[i] == z) s --;
```

Чему будет равно значение переменной s после выполнения данной программы, при любых значениях элементов массива?

   a) Количеству элементов массива A, больших первого элемента массива
   b) Количеству элементов массива A, не превосходящих первого элемента массива
   c) Количеству элементов массива A, не равных первому элементу массива
   d) Количеству элементов массива A, равных первому элементу массива

2. В программе описан одномерный целочисленный массив с индексами от 0 до n (т.е. первый элемент имеет индекс 0, последний - индекс n).

```
int s = 0;
for ( i = 1 ; i<= n ; i ++)
if(A[i] == A[0]) s = i;
```

Чему будет равно значение переменной s после выполнения данной программы?
Ответ должен быть верным при любых значениях элементов массива.

a) Минимальному элементу в массиве A
b) Количеству элементов массива A, равных первому элемента массива
c) Сумме всех элементов массива A, равных последнему элементу массива
d) Наибольшему индексу k, для которого элемент массива с индексом k равен первому элементу массива

3. В программе описан одномерный целочисленный массив с индексами от 0 до n. Ниже представлен фрагмент программы, обрабатывающей данный массив:

```
int s = 0,i;
int z = A[n];
for ( i = 0 ; i<= n-1 ; i ++)
if(A[i] > z) s = s + A[i];
```

Чему будет равно значение переменной s после выполнения данной программы?
Ответ должен быть верным при любых значениях элементов массива.

a) Максимальному элементу в массиве A
b) Количеству элементов массива A, больших последнего элемента массива
c) Сумме всех элементов массива A, больших последнего элемента массива
d) Индексу последнего элемента массива A, который больше A[n]

4. В программе описан одномерный целочисленный массив с индексами от 0 до n (т.е. первый элемент имеет индекс 0, последний - индекс n).

```
int i=1, s=0;
while(i<=n){
if(A[i]==A[0]) s=i;
i++;
}
```

Чему будет равно значение переменной s после выполнения данной программы?
Ответ должен быть верным при любых значениях элементов массива.

a) Минимальному элементу в массиве A
b) Количеству элементов массива A, равных первому элемента массива
c) Сумме всех элементов массива A, равных последнему элементу массива
d) Наибольшему индексу k, для которого элемент массива с индексом k равен первому элементу массива

5. Определите, что будет напечатано в результате выполнения программы

```
#include <stdio.h>
void main(){
int n=0, s=512;
while ( s >= 0){
s = s - 20;
n = n + 1;
```

```
}
printf (“%d”, n);
}
```

6. Определите, что будет напечатано в результате выполнения программы

```
#include <stdio.h>
void main(){
int n=24, s=0;
while ( n <= 28){
s = s + 20;
n = n + 2;
}
printf (“%d”, s);
}
```

7. Определите, что будет напечатано в результате выполнения программы

```
int n, s;
n=14;
s=0;
while (n<=18)
{
  s+=25;
  n++;
}
printf(“%d”, s);
```

8. Определите, что будет напечатано в результате выполнения программы

```
int s=0,n;
for(n=24;n<=28;n=n+2)
s+=20;
printf(“%d”,s);
```

9. Что будет напечатано в результате выполнения алгоритма

```
#include <stdio.h>
int F (int x){
return 19*(x-19)*(x+17)+17;
}
void main (){
int a, b, t, M, R;
a = -20; b=20;
M = a; R = F(a);
for (t = a ; t <= b; t++) {
    if (F(t) < R) {
        M = t; R = F(t);
    }
}
printf("%d", M);
}
```

10. Что будет напечатано в результате выполнения алгоритма

```
#include <stdio.h>
int F (int x){
return 19*(x-1)*(x-1);
}
void main (){
int a, b, t, M, R;
a = -10; b=20;
M = a; R = F(a);
for (t = a ; t <= b; t++) {
    if (F(t) > R) {
        M = t; R = F(t);
    }
}
printf("%d", M);
}
```

11. Что будет напечатано в результате выполнения алгоритма

```
#include <stdio.h>
int F (int x){
return 21-19*(x-1)*(x-1);
}
void main (){
int a, b, t, M, R;
a = -10; b=20;
M = a; R = F(a);
for (t = a ; t <= b; t++) {
if (F(t) > R) {
M = t; R = F(t);
}
}
printf("%d", R);
}
```

12. Что будет напечатано на экране

```
#include <stdio.h>
#define Sum(p) p + 2

void main () {
int var = 10 * Sum(5);
printf(“%d”, var);
}
```

13. Что будет напечатано на экране

```
#include <stdio.h>
#define square(x) x*x

void main () {
int z=5;
int var = square(z+1);
printf(“%d”, var);
```

```
}
```

14. Что будет напечатано на экране

```
#include <stdio.h>
#define Sum(p) (p + 2)*4+7

void main () {
int var = 10 +10* Sum(3);
printf(“%d”, var);
}
```

15. Что будет напечатано на экране

```
#include <stdio.h>
#define cube(x) x*x*(x)

void main () {
int z=2,var;
var = cube(z+3);
printf(“%d”, var);
}
```

16. Что будет напечатано на экране

```
#include <stdio.h>
#define cube(x) (x)*x*(x)

void main () {
int z=2,var;
var = cube(z+3);
printf(“%d”, var);
}
```

17. Что вы увидите на экране после выполнения следующего фрагмента

```
int ar[] = {-72, 3, 402, -1, 55, 132};
int* p = ar;
for (int i=101; i<=106; i++) {
 printf(“%d\n”, *p);
 p++;
}
```

18. Что вы увидите на экране после выполнения следующего фрагмента

```
int ar[] = {-72, 3, 402, -1, 55, 132};
int* p = ar;
for (int i=101; i<=106; i++) {
 printf(“%d\n”, *(p+5));
 p--;
}
```

19. Что вы увидите на экране после выполнения следующего фрагмента

```
int ar[] = {-72, 3, 402, -1, 55, 132};
int* p = ar;
for (int i=101; i<=106; i++) {
 printf(“%d”, *p);
 p++;
}
```

20. Что вы увидите на экране после выполнения следующего фрагмента

```
int ar[] = {-72, 3, 402, -1, 55, 132};
for (int i=0; i<=4; i++)
        printf(“%d”, *(ar+i));
```

21. Что вы увидите на экране после выполнения следующего фрагмента

```
int array[] = {3, 4, 1007, 8, 5, 12};
int *x;
x=array;
for (int i=0; i<=4; i++)
        printf(“%d”, *(x+i));
```

22. Что вы увидите на экране после выполнения следующего фрагмента

```
int ar[20] = {4, 0, 13, 21, 5, 120, 5, 17,88};
int* p = ar+5;
for (int i=10; i<=13; i++){
 printf(“%d”, *p+2);
 p++;
}
```

23. Написать программу, позволяющую найти и вывести максимальное значение среди не кратных 9 трехзначных элементов целочисленного массива, заполненного случайными числами в диапазоне от 100 до 10000. Если таковых не будет найдено - вывести сообщение.
24. Дан массив из 10 элементов: 2, 104, 32, 124, 17, 65, 33, 87, 22, 45. Написать программу вывода на экран элементов массива, больших заданного числа **x**, и их количества. Доступ к элементам массива получать через указатель.
25. Написать программу, позволяющую определить из каких цифр состоит заданное натуральное трехзначное число и выводящую на экран эти цифры в порядке возрастания.
26. Дан двумерный массив размера [3][5]. Определить сколько ненулевых элементов в массиве.
27. Написать программу, позволяющую заменить все элементы целочисленного массива из 15 элементов, стоящие до максимального нулями.
28. Написать функцию для нахождения длины отрезка по введенным координатам его концов. Координаты концов отрезка задать с помощью структур.