

## MIDTERM 1 CS111 Fall 2019

NAME \_\_\_\_\_ CUNYID \_\_\_\_\_

Instructions:

- 1) Make sure your name and CUNY ID are filled in.
- 2) When asked to write a program, begin with the main function of the program. In most cases I have written the beginning of the function for you and you just fill in the rest.

### SECTION 1 – General Questions 5 Points each

Q1) Which functions below have incorrect syntax? Also note what is the error. E.g. q) missing “;”

- a) void F1(){ }
- b) void 8F(){ }
- c) double D1(){ double d = 9; }
- d) int L1(){return(1)}
- e) char L2{return('C');}

Q1) b) Function name cannot start with number. c) no value returned. d) missing “;”. e) missing parentheses.

Q2) TRUE/FALSE

**C/C++ has a special feature in that the user has direct access to computer memory.**

Q2) TRUE

Q3) TRUE/FALSE

**The C/C++ compiler translates C/C++ code into C# and then it is linked into Java.**

Q3) FALSE – Links to machine code

Q4) **How come a char can only contain a number up to 255 while an integer (on most computers) can store values over 1 billion.**

Q4) char is size one byte (2^8) while an integer is 4 bytes (2^32)

Q5) Complete the truth table below. Enter T or F in the third column.

TRUTH TABLE		
A	B	NOT(A AND B)
T	T	F
T	F	T
F	T	T
F	F	T

Q6) In the case below where i is declared as an integer, What is another way to write  $i = i + 1$ ?

Q6) i++\_\_\_\_\_

7) What is the purpose of the srand function?

Q7) To stimulate random we seed the a random function with a random value (like the current time)

8) int, char, double, and decimal are C/C++ data types. True or False?

FALSE - decimal is not a native C/C++ type

9) An array can be thought of as a bunch of variables of the same type one after the other. True or False? TRUE

10) integers can only store positive whole numbers. True or False? FALSE

**SECTION 2 – Logic 5 Points each**

What is the output?

Q1)

```
bool A() { return(false); }
if('p' && A() && 1/0)
    cout << "Condition Met";
else
    cout << "Condition Not Met";
```

Q1) "Condition Not Met" or "divide by zero"

Q2)

```
if (!(true && true) || !true)
    cout << "condition true";
else
    cout << "condition not true";
```

Q2) "condition not true"

Q3)

```
if (5 && 4 && 5.0/7)
    cout << "condition true";
else
    cout << "condition not true";
```

Q3) "condition true"

Q4)

```
if(6/7>1||6%2>1||120%10>1||123/100>1)
    cout << "condition true";
else
    cout << "condition not true";
```

Q4) "condition not true"

Q5)

```
int i = 1.234;
double d = 1.234 - i;
d++;
i--;
if(i>0|| d>1)
    cout << "condition true";
else
    cout << "condition not true";
```

Q5) condition true

## Section 3 – Understanding Programming Logic 2 Points Each

Q1) Extra Credit - What pattern does the function bubble() output?

It prints the numbers in ascending order.

```
void Bubble()
{
    int a=rand(), b=rand(), c=rand();

    if (a > b)
    {
        int temp = a;
        a = b;
        b = temp;
    }

    if (b > c)
    {
        int temp = b;
        b = c;
        c = temp;
    }
    if (a > b)
    {
        int temp = a;
        a = b;
        b = temp;
    }
    cout << a << b << c << endl;
}
```

Q2) Extra Credit - What pattern does the function M() output?

It returns the Maximum or Largest number

```
int M()
{
    int a, b, c;
    int a=rand(), b=rand(), c=rand();
    int m = a;

    if (m < b) m = b;

    if (m < c) m = c;

    return(m);
}
```

## Section 4 – Programming 10 Points each

### Q1) Random double number

Write a program that asks the user to enter two double numbers up to three decimal places the program then outputs a random number between the numbers entered. For example, if the user enters 2.123 and 7.890 then return a random number between 2.123 and 7.890

```
int main()
{
    cout << "please enter two decimal numbers" << endl;
    cout << " in ascending order, first the smaller" << endl;
    cout << " and then the larger one." << endl;
    cout << "E.G. 1.456 and 5.890" << endl;
    cout << "I will return a random number between those two" << endl;
    double a, b;
    cin >> a >> b;
    a *= 1000;
    b *= 1000;
    int aa = a, bb = b;
    srand(time(0));
    // note: entering 1.111 and 1.112 will cause
    // a divide by 0
    double answer = aa + 1 + (rand() % (bb - aa - 1));
    cout << answer / 1000 << endl;
    return(0);
}
```

## Q2) Leap Year

Write a program that tells you whether an English year is a leap year.

The formula is:

Year is multiple of 400.

Year is multiple of 4 and not multiple of 100.

Examples:

2004 → Leap Year; 1900 → NOT Leap year; 2000 → Leap year

```
int main()
{
    cout << "Enter a year and I will tell you if it ";
    cout << "is a leap year." << endl;
    int year;
    cin >> year;

    // method one
    if((year%400==0)|| ((year%4==0) && (year%100)))
        cout << "This is a leap year" << endl;
    else
        cout << "This is NOT a leap year" << endl;

    // method two
    if (year % 400 == 0)
        cout << "This is a leap year" << endl;
    else if(year % 100 == 0)
        cout << "This is NOT a leap year" << endl;
    else if (year % 4 == 0)
        cout << "This is a leap year" << endl;
    else
        cout << "This is NOT a leap year" << endl;

    return(0);
}
```

## ASCII CHART

Dec	Hex	Oct	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr
0	0	000	NULL	32	20	040	&#032;	Space	64	40	100	&#064;	@	96	60	140	&#096;	`
1	1	001	Start of Header	33	21	041	&#033;	!	65	41	101	&#065;	A	97	61	141	&#097;	a
2	2	002	Start of Text	34	22	042	&#034;	"	66	42	102	&#066;	B	98	62	142	&#098;	b
3	3	003	End of Text	35	23	043	&#035;	#	67	43	103	&#067;	C	99	63	143	&#099;	c
4	4	004	End of Transmission	36	24	044	&#036;	\$	68	44	104	&#068;	D	100	64	144	&#100;	d
5	5	005	Enquiry	37	25	045	&#037;	%	69	45	105	&#069;	E	101	65	145	&#101;	e
6	6	006	Acknowledgment	38	26	046	&#038;	&	70	46	106	&#070;	F	102	66	146	&#102;	f
7	7	007	Bell	39	27	047	&#039;	'	71	47	107	&#071;	G	103	67	147	&#103;	g
8	8	010	Backspace	40	28	050	&#040;	(	72	48	110	&#072;	H	104	68	150	&#104;	h
9	9	011	Horizontal Tab	41	29	051	&#041;	)	73	49	111	&#073;	I	105	69	151	&#105;	i
10	A	012	Line feed	42	2A	052	&#042;	*	74	4A	112	&#074;	J	106	6A	152	&#106;	j
11	B	013	Vertical Tab	43	2B	053	&#043;	+	75	4B	113	&#075;	K	107	6B	153	&#107;	k
12	C	014	Form feed	44	2C	054	&#044;	,	76	4C	114	&#076;	L	108	6C	154	&#108;	l
13	D	015	Carriage return	45	2D	055	&#045;	-	77	4D	115	&#077;	M	109	6D	155	&#109;	m
14	E	016	Shift Out	46	2E	056	&#046;	.	78	4E	116	&#078;	N	110	6E	156	&#110;	n
15	F	017	Shift In	47	2F	057	&#047;	/	79	4F	117	&#079;	O	111	6F	157	&#111;	o
16	10	020	Data Link Escape	48	30	060	&#048;	0	80	50	120	&#080;	P	112	70	160	&#112;	p
17	11	021	Device Control 1	49	31	061	&#049;	1	81	51	121	&#081;	Q	113	71	161	&#113;	q
18	12	022	Device Control 2	50	32	062	&#050;	2	82	52	122	&#082;	R	114	72	162	&#114;	r
19	13	023	Device Control 3	51	33	063	&#051;	3	83	53	123	&#083;	S	115	73	163	&#115;	s
20	14	024	Device Control 4	52	34	064	&#052;	4	84	54	124	&#084;	T	116	74	164	&#116;	t
21	15	025	Negative Ack.	53	35	065	&#053;	5	85	55	125	&#085;	U	117	75	165	&#117;	u
22	16	026	Synchronous idle	54	36	066	&#054;	6	86	56	126	&#086;	V	118	76	166	&#118;	v
23	17	027	End of Trans. Block	55	37	067	&#055;	7	87	57	127	&#087;	W	119	77	167	&#119;	w
24	18	030	Cancel	56	38	070	&#056;	8	88	58	130	&#088;	X	120	78	170	&#120;	x
25	19	031	End of Medium	57	39	071	&#057;	9	89	59	131	&#089;	Y	121	79	171	&#121;	y
26	1A	032	Substitute	58	3A	072	&#058;	:	90	5A	132	&#090;	Z	122	7A	172	&#122;	z
27	1B	033	Escape	59	3B	073	&#059;	;	91	5B	133	&#091;	[	123	7B	173	&#123;	{
28	1C	034	File Separator	60	3C	074	&#060;	<	92	5C	134	&#092;	\	124	7C	174	&#124;	
29	1D	035	Group Separator	61	3D	075	&#061;	=	93	5D	135	&#093;	]	125	7D	175	&#125;	}
30	1E	036	Record Separator	62	3E	076	&#062;	>	94	5E	136	&#094;	^	126	7E	176	&#126;	~
31	1F	037	Unit Separator	63	3F	077	&#063;	?	95	5F	137	&#095;	_	127	7F	177	&#127;	Del

asciicharstable.com