Name: Key Section: 

Instructions:

1. There are test questions on the front and the back of each sheet.

2. This is a closed book exam. Do not use any notes, books, or neighbors except your one page, two side, handwritten, cheat sheet which MUST be turned in with your exam.

3. Show your work. Partial credit will be given. Grading will be based on correctness and clarity.

4. You have 75 minutes to complete the exam. Watch your time appropriately. You should take about 15 minutes per question section.

Integrity: The University of Richmond’s Honor Code is “We, the students of the University of Richmond, shall promote and uphold a community of integrity and trust.” Upon accepting admission to University of Richmond, a student immediately assumes a commitment to uphold the Honor Code, to accept responsibility for learning, and to follow the philosophy and rules of the Honor System. Students will be required to state their commitment on examinations, research papers, and other academic work. Ignorance of the rules does not exclude any member of the Richmond community from the requirements or the processes of the Honor System.

I agree to uphold this commitment and produce original work in this exam, i.e., I will not cheat nor will I consciously let anyone cheat.

Signature: ________________________________

DO NOT BEGIN THE EXAM UNTIL INSTRUCTED TO DO SO. GOOD LUCK!

<table>
<thead>
<tr>
<th>Question</th>
<th>Points received</th>
<th>possible</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>Bonus</td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>
1. **Definitions** (20 points, 2 points each). For each of the following, fill in the blanks with the most accurate term.

(a) Computer science is the study of computer architecture, problem solving, and interaction with computing devices.

(b) There are numerous types of errors that may occur in computing. An error reported by the compiler when it sees invalid code is called a syntax error, and an error that occurs from encountering an impossible command while running a program is called an exception/runtime error.

(c) Computer programs execute while interacting with both the operating system and user.

(d) The data type defines both the possible values of the data and the operations on the data.

(e) An expression is any combination of variables, operators, and function calls that generates a new value.

(f) Type conversion is the process of reinterpreting the value of a variable as a new data type. It can happen in two ways, explicitly with a cast or implicitly by Java.

(g) The sequence of statements that the program executes is called the control flow. It also refers to the possible paths the program could take.

(h) Nesting occurs when a control structure is located within another control structure.

(i) Scope is the set of rules defining where in a program you are allowed to refer to a variable.

(j) Methods are subroutines that provide organization and re-usability to portions of programs. Input is passed to the subroutine during invocation and is referred to as the arguments.
2. **Syntax** (20 points). The following question deal specifically with Java syntax. Always write snippets of code in valid Java. Don’t forget your semicolons (where appropriate).

(a) Write the signature of the `main` method (2 points):

\[
public \ static \ void \ main(String[] \ args)
\]

(b) Given the code: `double avg = 5.3;`. Write the snippets for the following (6 points):

<table>
<thead>
<tr>
<th>Type</th>
<th><code>double</code></th>
</tr>
</thead>
<tbody>
<tr>
<td>Declaration</td>
<td><code>double avg</code></td>
</tr>
<tr>
<td>Identifier</td>
<td><code>avg</code></td>
</tr>
<tr>
<td>Literal</td>
<td><code>5.3</code></td>
</tr>
<tr>
<td>Assignment</td>
<td><code>avg = 5.3</code></td>
</tr>
<tr>
<td>Initialization</td>
<td><code>avg = 5.3</code></td>
</tr>
</tbody>
</table>

(c) Given that `a` is a `short`, `b` is an `int`, and `c` is a `float`, what is the type of the expression `a*b + c`? (2 points)

```
float
```

(d) Write a snippet of code to convert a `double d` to a `long l` (2 points).

```
long l = (long)d;
```

(e) Write a snippet of code to randomly generate an integer `c` in the range `[a, b]` where `a` is an integer and `b` is an integer (2 points). You must use `Math.random()`.

```
int c = (int)(Math.random()*(b - a + 1)) + a;
```

(f) What is the Java keyword to immediately skip to the condition check of a loop (2 points)?

```
continue
```

(g) Let the variable `in` be a `Scanner`. Write a snippet of Java code to initialize a variable `s` of type `short` (2 points).

```
short s = in.nextShort();
```

(h) What is the Java keyword to declare that a variable’s value is not allowed to be altered (2 points)?

```
final
```
3. **Tracing** (30 points, 5 points each). For each of the following pieces of pseudocode or Java, determine the final output. No partial credit will be given in this section. You should assume there are no compiler errors, however, there may be runtime exceptions and infinite loops. If an infinite loop occurs write **Does not terminate**, and if an exception occurs write **Error**: followed by the Java error that is reported as closely as possible (I will be lenient here). Carefully, work through each example one step at a time.

(a) ```java
int x = 3;
int y = x;
x = 3*y;
y = x/2;
x = x + 1;
System.out.println(y+x);
```
Output: 14

(b) ```java
Integer x ← 23
if x % 7 = 3 then
   x ← x * 2
else if x % 4 = 3 then
   x ← x / 2
Output(x);
```
Output: 11

(c) ```java
Real number x ← 4
Real number y ← −4
while x ≥ 0.25 do
   x ← x / 2
   y ← y + x
Output(x + y)
```
Output: 0

(d) ```java
String s = "aRRiMaPiRaTe";
int c = 0;
for(int a = s.length(); a >= 1; --a) {
   if(s.charAt(s.length() − a) == s.charAt(s.length() − a + 1))
      c += 5;
else
   c += 1;
}
System.out.println(c);
```
Output: Error: Invalid index
(e) byte b = 1;
do {
    b *= 2;
} while(b < 128)
System.out.println(b);

Output: ___________ Does not terminate

(f) int k = 7;
    int m = 14;
    for(int l = 0; l < k; ++l)
        for(int s = k; s > l; --s)
            ++m;
    System.out.println(m);

Output: _______________ 42 _______________
4. **Diagramming** (20 points, 5 points each). For each of the following create a flow chart (circles, blocks, diamonds, arrows, and labels), but do not describe the individual elements. Simply label items appropriately.

(a) Control flow of a **for**-loop.

(b) **Generic algorithm**.

(c) High level architecture of computer hardware.

(d) Control flow of a **do-while**-loop.
5. Algorithm (10 points). For the following question, please write pseudocode (not Java code) for a function.

Write an algorithm that finds all of the factors of an integer. Assume you are given as input an integer \( N \), then your algorithm should output all of the smallest factors of \( N \) in increasing order. For example, if \( N = 120 \) then the algorithm will output 2 2 2 3 5.

Function Factors
Input: Integer \( N \)
Integer \( f \) ← 2
while \( N > 1 \) do
    if \( N \% f = 0 \) then
        \( N \leftarrow N / f \)
        Output(\( f \))
    else
        \( f \leftarrow f + 1 \)
6. **Bonus** (up to 10 points, 5 points each). Answer the following conceptual or programming questions on computing and Java.

(a) Considering `System.out.printf`, how do you output a number in scientific notation with 4 significant digits? How do you output a number in hexadecimal (base 16) format?

\[
\text{\%e} \\
\text{\%x}
\]

(b) Write a valid `switch` statement for handling a keyboard press to move a point. Assume the key pressed is stored in a `char` named `key` and the point has an `x` and `y` value. When ‘a’, ‘s’, ‘d’, or ‘w’ is typed move the point by 1 unit in an appropriate direction. If any other key is pressed reset the point to the origin.

```java
switch(key) {
    case 'a': x -= 1; break;
    case 's': y -= 1; break;
    case 'd': x += 1; break;
    case 'w': y += 1; break;
    default: x = y = 0; break;
}
```