Last name(s) **Name** **ID**

Midterm EDA Exam Length: 2.5 hours 06/11/2017

- *The exam has 4 sheets, 8 sides and 4 problems.*
- *Write your full name and ID on every sheet.*
- *Write your answers to all problems in the exam sheets within the reserved space.*
- *Unless otherwise indicated, all your answers must be justified.*
- *As usual, if the base of logarithms is not explicit, the base is 2.*

Problem 1 (3 points)

Answer the following questions. You do not need to justify your answers.

(a) (0.5 pts.) The cost in time of the following code snippet as a function of n:

```
int j = 0;
int s = 0;
for (int i = 0; i < n; ++i)
  if (i == j*j) {
    for (int k = 0; k < n; ++k) ++s;
    ++j;
  }
```

is $\Theta($ $)$.

(b) (1 pt.) Given a vector of integers v and an integer x, the function

```
int position (const vector<int>& v, int x) {
  int n = v.size ();
  for (int i = 0; i < n; ++i)
    if (v[i] == x)
      return i;
  return −1;
}
```

examines the $n = v.size()$ positions of v and returns the first one that contains x, or -1 if there is none.

Assume that x occurs in vector v. The asymptotic cost in time of *position* in the worst case is $\Theta($ $)$, and in the average case (assuming uniform probability) is $\Theta($ $)$.

(c) (0.5 pts.) Give the order of magnitude of the following function in its simplest form: $5(\log n)^2 + 2\sqrt{n} + \cos(n^8) = \Theta($ $)$.

(d) (0.5 pts.) The solution to the recurrence

$$T(n) = \begin{cases} 1 & \text{if } 0 \leq n < 2 \\ 4 \cdot T(n/2) + 3n^2 + 2\log(\log n) + 1, & \text{if } n \geq 2 \end{cases}$$

is $T(n) = \Theta(\boxed{})$.

(e) (0.5 pts.) The solution to the recurrence

$$T(n) = \begin{cases} 1 & \text{if } 0 \leq n < 2 \\ 4 \cdot T(n-2) + 3n^2 + 2\log(\log n) + 1, & \text{if } n \geq 2 \end{cases}$$

is $T(n) = \Theta(\boxed{})$.

Last name(s) **Name** **ID**

Problem 2 **(2.5 points)**

Consider the following program, which reads a strictly positive integer m and a sequence of n integers $a_0, a_1, \ldots, a_{n-1}$ that are guaranteed to be between 1 and m:

```
int main() {

  int m;
  cin >> m;
  vector<int> a;
  int x;
  while (cin >> x)
    a.push_back(x);

  vector<int> b(m + 1, 0);
  int n = a.size();
  for (int i = 0; i < n; ++i)
    ++b[ a[i] ];

  for (int j = 1; j <= m; ++j)
    b[j] += b[j-1];
}
```

(a) (0.5 pts.) Given y such that $0 \leq y \leq m$, what is the value of $b[y]$ at the end of *main* in terms of the sequence a?

(b) (1 pt.) Let us define the *median* of the sequence a as the value p between 1 and m such that $b[p] \geq \frac{n}{2}$ and $b[p-1] < \frac{n}{2}$, where b is the vector computed as in the code above.

Write in C++ a function

```
int median(int n, const vector<int>& b);
```

which, given n, the size of the sequence a, and b, computed as above, finds the median of a. Solutions with cost $\Omega(m)$ will not be considered as valid answers. If you use auxiliary functions, implement them too.

(c) (1 pt.) Analyze the cost of your function *median* of the previous exercise as a function of m.

Last name(s) **Name** **ID**

Problem 3 **(2 points)**

Consider the type

```
struct complex {
  int real ;
  int imag;
};
```

for implementing complex numbers (with integer components), where *real* is the real part and *imag* is the imaginary part of the complex number.

For example, if z is an object of type *complex* that represents the number $2 + 3i$, then $z.real = 2$ and $z.imag = 3$.

(a) (1 pt.) Implement in C++ a function

```
complex exp(complex z, int n);
```

which, given a complex number z and an integer $n \geq 0$, computes the complex number z^n. The solution must have cost $\Theta(\log n)$ in time. If you use auxiliary functions, implement them too.

Note. Recall that the product of complex numbers is defined as follows:

$$(a + bi) \cdot (c + di) = (ac - bd) + (ad + bc)i$$

(b) (1 pt.) Justify that the cost in time of your function *exp* is $\Theta(\log n)$.

Last name(s) **Name** **ID**

Problem 4 **(2.5 points)**

Consider the following recurrence:

$$T(n) = T(n/2) + \log n$$

(a) (1 pt.) Let us define the function U as $U(m) = T(2^m)$. Using the recurrence for T, deduce a recurrence for U.

(b) (0.5 pts.) Solve asymptotically the recurrence for U of your answer to the previous exercise.

(c) (1 pt.) Solve asymptotically the recurrence for T.

This side would be intentionally blank if it were not for this note.