

Problem 1: Runtime Analysis

Part (a)

void f1(int n)

{

 int i=2; $\Leftarrow O(1)$

 while ($i \leq n$) {

 /* do something that takes $O(1)$ time */

$i = i * i;$

} 3

$$i=2 \quad i=2^2 \quad i=4^2 \quad i=16^2$$

$$i=2, 4, 16, 256$$

Find
how many iterations $\Rightarrow i = 2^{(2^k)}$

$$2^{(2^k)} \geq n$$

$$2^k \geq \log_2(n)$$

$$k \geq \log_2(\log_2(n))$$

$$= \Theta(\log_2(\log_2(n)))$$

Part (b)

void f2(int n)

{

 for (int i=1; i<=n; i++) {

 if ((i % (int)sqrt(n)) == 0) {

 for (int k=0; k< pow(i, 3); k++) {

 /* do something that takes $O(1)$ time */

} 3

} 3

$$T(n) = \sum_{k=1}^{\sqrt{n}} (k \cdot \sqrt{n})^3$$

$$T(n) = k^3 \cdot \sqrt{n}^3$$

$$T(n) = n^{3/2} \cdot \frac{n^3}{4}$$

Outer:

$$i \% \sqrt{n} == 0$$

$$i = K \cdot \sqrt{n}$$

$$K \cdot \sqrt{n} \leq n$$

$$K \leq \sqrt{n}$$

Inner:

$$K < i^3$$

$$\sum_{k=1}^m k^3 = \left(\frac{m \cdot (m+1)}{2}\right)^2$$

$$= \left(\frac{\sqrt{n} \cdot (\sqrt{n}+1)}{2}\right)^2$$

$$= \frac{(\sqrt{n} \cdot \sqrt{n})^2}{4}$$

Part (c)

for (int i=1; i<=n; i++) {

executes n times

 for (int k=1; k<=n; k++) {

also n times

 if (A[k] == i) {

Count:

$$m=1, 2, 4, 8, 16, \dots n$$

 for (int m=1; m<=n; m=m+m) {

runs $O(\log_2(n))$

 // do something that takes $O(1)$ time

 // Assume the contents of the A[] array are not changed

$$n^2$$

} 3

} 3

$$T(n) = \sum_{i=1}^n \left(\sum_{k=1}^n O(\log_2(n)) \right) \Rightarrow O(\log_2(n)) \cdot n^2$$

$$T(n) = O(n^2 \cdot \log_2(n))$$

Part (d)

$\text{int } f(\text{int } n)$

{

O(1)

```
int *a = new int[10];
int size = 10;
for(int i=0; i<n; i++)
{
```

```
    if(i == size)
{
```

}

```
    int newsize = 3 * size / 2;
```

```
    int *b = new int[newsize];
```

```
    for(int j=0; j<size; j++) b[j] = a[j];
```

O(1)

```
    delete [] a;
```

```
    a = b;
```

```
    size = newsize;
```

}

```
    a[i] = i * i;
```

}

3

Size = 10

$3 \cdot 10/2$

$3 \cdot 15/2$

$3 \cdot 22/2$

newsize = 15, 22, 33

$10 \left(\frac{3}{2}\right)^k \geq n$

$\left(\frac{3}{2}\right)^k \geq \frac{n}{10}$

$k \geq \log_{3/2} \left(\frac{n}{10}\right)$

$O(\log n)$

$$\sum_{n=0}^{\log_{3/2}(n)} \left(\frac{3}{2}\right)^k = O(n) + O(1) + O(1)$$

$T(n) = O(n)$

Problem 2: Linked List Recursion Tracing

Struct Node {

```
    int val;
```

```
    Node* next;
```

};

```
Node* llrec(Node* in1, Node* in2)
```

{

```
    if(in1 == nullptr)
```

```
        return in2;
```

}

```
else if(in2 == nullptr)
```

```
    return in1;
```

}

```
else
```

```
    in1->next = llrec(in2, in1->next);
```

```
    return in1;
```

}

3

a)

go

backwards

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

1