**알고리즘 과제 #2**

201723274 소프트웨어학과 이서현

1. [40점] Bb 있는 Divide-and-Conquer 강의노트 (동영상을 말하는 것이 아니라 파워포인트 파일을 말함)의 62쪽과 63쪽에 각각 행렬 곱셈하는 divide-and-conquer 알고리즘이 쓰여 있다 (교재에서 스캔한 것이 아니라 내가 작성한 것). 63쪽의 Strassen’s algorithm에서 구하는 C11, C12, C21, C22이 각각 62쪽의 알고리즘이 구하는 C11, C12, C21, C22와 일치함을 보이시오.

M1 = MULT(ADD(A11,A22),ADD(B11,B22))

= (A11+A22)\*(B11+B22)

= A11\*B11+A11\*B22+A22\*B11+A22\*B22

M2 = MULT(ADD(A21,A22),B11)

= (A21+A22)\*B11

= A21\*B11+A22\*B11

M3 = MULT(A11,ADD(B12,-B22))

= A11\*(B12+(-B22))

= A11\*B12-A11\*B22

M4 = MULT(A22,ADD(B21,-B11))

= A22\*(B21-B11)

= A22\*B21-A22\*B11

M5 = MULT(ADD(A11,A12),B22)

= (A11+A12)\*B22

= A11\*B22+A12\*B22

M6 = MULT(ADD(A21,-A11),ADD(B11,B12))

= (A21-A11)\*(B11+B12)

= A21\*B11+A21\*B12-A11\*B11-A11\*B12

M7 = MULT(ADD(A12,-A22),ADD(B21,B22))

= (A12-A22)\*(B21+B22)

= A12\*B21+A12\*B22-A22\*B21-A22\*B22

C11 = ADD(M1, M4, -M5, M7)

= (A11\*B11+A11\*B22+A22\*B11+A22\*B22)+(A22\*B21-A22\*B11)

-(A11\*B22+A12\*B22)+( A12\*B21+A12\*B22-A22\*B21-A22\*B22)

= A11\*B11+A12\*B21

= ADD(MULT(A11,B11),MULT(A12,B21))

C12 = ADD(M3, M5)

= (A11\*B12-A11\*B22)+(A11\*B22+A12\*B22)

= A11\*B22+A12\*B22

= ADD(MULT(A11,B12),MULT(A12,B22))

C21 = ADD(M2,M4)

= (A21\*B11+A22\*B11)+(A22\*B21-A22\*B11)

= A21\*B11+A22\*B21

= ADD(MULT(A21,B11),MULT(A22,B21))

C22 = ADD(M1, M3, -M2, M6)

= (A11\*B11+A11\*B22+A22\*B11+A22\*B22)+(A11\*B12-A11\*B22)

-(A21\*B11+A22\*B11)+(A21\*B11+A21\*B12-A11\*B11-A11\*B12)

= A22\*B22+A21\*B12

= ADD(MULT(A21,B12),MULT(A22,B22))

1. [20점] (가) Dimension이 64 × 64 인 행렬 두 개를 standard algorithm 으로 곱할 때 실행되는 scalar multiplication의 횟수를 쓰시오. 답안을 642, 64\*64, log64 등의 형태로 놔두지 말고 끝까지 계산해서 단순한 정수 한 개로 표시해야 함. 답안을 642, 64\*64, log64 등의 형태로 쓴 경우에는 0점으로 처리함. Standard algorithm이란 교재의 제 9 쪽에 있는 Algorithm 1.4를 뜻함.

T(n)=8\*T(n/2)

T(n)=a\*T(n/b)+cd 에서 a=8, b=2 이므로

T(n)=nlog28= n3이다.

따라서 64x64 matrix이므로 643인 262144이다.

(나) Dimension이 64 × 64 인 행렬 두 개를 standard algorithm 으로 곱할 때 실행되는 scalar addition의 횟수를 쓰시오. 답안을 642, 64\*64, log64 등의 형태로 놔두지 말고 끝까지 계산해서 단순한 정수 한 개로 표시해야 함. 답안을 642, 64\*64, log64 등의 형태로 쓴 경우에는 0점으로 처리함. Standard algorithm이란 교재의 제 9 쪽에 있는 Algorithm 1.4를 뜻함.

Scalar addition은 T(n)=n3-n2번 실행되므로

643-642=258048 이다.

1. [20점] (가) Dimension이 64 × 64 인 행렬 두 개를 Strassen’s algorithm 으로 곱할 때 실행되는 scalar multiplication의 횟수를 쓰시오. 답안을 642, 64\*64, log64 등의 형태로 놔두지 말고 끝까지 계산해서 단순한 정수 한 개로 표시해야 함. 답안을 642, 64\*64, log64 등의 형태로 쓴 경우에는 0점으로 처리함.

T(n)=7T(n/2), T(1)=1이다.

T(2)=7T(2/2)=7T(1)=7

T(4)=7T(4/2)=7T(2)=72

…

T(n)=7lg n=nlg 7≈n2.81

따라서 T(64)=642.81=118950.35072573094 이다.

(나) Dimension이 64 × 64 인 행렬 두 개를 Strassen’s algorithm 으로 곱할 때 실행되는 scalar addition의 횟수를 쓰시오. 답안을 642, 64\*64, log64 등의 형태로 놔두지 말고 끝까지 계산해서 단순한 정수 한 개로 표시해야 함. 답안을 642, 64\*64, log64 등의 형태로 쓴 경우에는 0점으로 처리함.

T(n)=7T(n/2)+18(n/2)2 T(1)=0

tk=T(2k)라 하면

tk=7tk-1+18(2k-1)2

=7k-1+4k(18/4)

=c17k+c24k

T(2k)=c17k+c24k

T(n)=c17lg n+c24lg n

=c1nlg7+c2n2

T(n)=6nlg7-6n2≈6n2.81-6n2 이므로

T(64)=6\*642.81-6\*642=689126.104354 이다.

1. [20점] 교재 제 75쪽 Analysis of Algorithm 2.8에서 구한 recurrence relation을 푸시오. Asymptotic class를 구하는 것이 아니라 을 에 관한 식(formula)으로 정확하게 표현해야 함.

이 recurrence relation의 풀이가 교재의 Appendix B Example 20에 나와 있다. 그러나, 이 풀이를 베끼지 말고, 반복적인 대입(repetitive substitution) 만을 사용하여 recurrence relation을 풀어야 함.

T(n)=7T(n/2)+18(n/2)2

=7(7T(n/4)+18(n/4)2)+18(n/2)2 = 72T(n/4)+7\*18(n/4)2+18(n/2)2

=72(7T(n/8)+18(n/8)2)+7\*18(n/4)2+18(n/2)2=73T(n/8)+72\*18(n/8)2+7\*18(n/4)2+18(n/2)2

=…

=7lgnT(n/n)+18(7lgn-1(n2/4lgn)+…+(n2/22)) *T(1)=0*

=18n2(1+(7/4)+…+(7/4)lgn-1)/4

=(18n2)/4

= \* =6n2(nlg(7/4)-1)

=6nlg7-6n2

≈6n2.81-6n2 ∈ Θ(n2.81)

1. [20점] (가) 교재의 제 86쪽에 있는 Mergesort algorithm의 최적의 threshold t를 구하는 부등식 (2.7)을 푸시오. 교재에는 식 (2.7)이 등식으로 표현되어 있으나, 이 것을 (좌변≥우변) 형태의 부등식으로 간주하여 푸시오. Threshold t 가 홀수인 경우와 짝수인 경우로 나누어 풀어야 함. 답만 쓰면 안되고 풀이 과정을 보여야 함. 답만 쓰면 0점으로 처리함.

* t가 짝수인 경우

((t/2)(t/2-1))/2+((t/2)(t/2-1))/2+32t ≥ t(t-1)/2

(t/2)(t/2-1)+32t ≥ t(t-1)/2

t2-128t ≤ 0

t ≤128

* t가 홀수인 경우

=t-1이고 이다.

++32t ≥ t(t-1)/2

(t-1)(t-3)+(t+1)(t-1)+256t ≥ 4t(t-1)

t2-128t-1≤0

t ≤ 128.0078

(나) 교재의 제 87쪽에 있는 large integer multiplication algorithm의 최적의 threshold t를 구하는 부등식을 푸시오. 교재에는 등식으로 표현되어 있으나, 이 것을 (좌변≥우변) 형태의 부등식으로 간주하여 푸시오. Threshold t 가 홀수인 경우와 짝수인 경우로 나누어 풀어야 함. 답만 쓰면 안되고 풀이 과정을 보여야 함. 답만 쓰면 0점으로 처리함.

* t가 짝수인 경우

3(t/2)2+16t ≥ t2

3t2+64t ≥ 4t2

t2-64t ≤ 0

t ≤ 64

* t가 홀수인 경우

3((t+1)/2)2+16t ≥ t2

3(t+1)2+64t ≥ 4t2

4t2-3(t+1)2-64t ≤ 0

t2-70t-3 ≤ 0

t ≤ 70.04