

1 Instruction

1.1 Definiton of Instruction

Define instruction s

$$\begin{aligned} & \text{instruction } s := \\ & (s \equiv \text{function}) \oplus (s \equiv \text{operation}) \end{aligned}$$

2 Instruction Set

2.1 Definiton of Instruction Set

Define \mathcal{I} ; an ordered set of computational instructions s_i

$$\mathcal{I} := \{s_1, s_2, \dots, s_N\}$$

2.2 Abstraction Notation

Define \mathcal{I} ; an ordered set of computational instructions s_i

$$\begin{aligned} & \text{"Instruction Set"} := \mathcal{I} : \\ & (\mathcal{I} \equiv \text{Set}) \wedge (s_i \equiv \text{instruction} \quad \forall s_i \in \mathcal{I}) \end{aligned}$$

3 Memory

3.1 Definiton of Memory

$$\begin{aligned} \text{memory } m &:= \\ m &\in \Omega \end{aligned}$$

3.2 Definition of Memory Set

Define Memory \mathcal{M} ; an ordered set of objects m_i

$$\mathcal{M} := \{m_1, m_2, \dots, m_M\}$$

3.3 Abstraction Notation

Define Memory \mathcal{M} ; an ordered set of either bools numbers or objects m_i

$$\begin{aligned} \text{"Memory Set"} &:= \mathcal{M} : \\ (\mathcal{M} \equiv \text{Set}) \wedge (m_i \equiv \text{object } \forall m_i \in \mathcal{M}) \end{aligned}$$

4 Program

4.1 Definition of a Program

Define program \mathcal{P} ; a tensor of instructions and memory

$$\begin{aligned}\mathcal{P} &:= \{\{s_1, s_2, \dots, s_N\}, \{m_1, m_2, \dots, m_M\}\} \\ &= \langle \mathcal{I}, \mathcal{M} \rangle\end{aligned}$$

4.2 Abstraction Notation

Define a method \mathcal{P} ; a tensor of instructions and memory

$$\begin{aligned}\text{"Method"} &:= \mathcal{P} : \\ (\mathcal{P} \equiv \text{Tensor}) &\wedge (\mathcal{P} \supseteq \mathcal{I}) \wedge (\mathcal{P} \supseteq \mathcal{M})\end{aligned}$$

4.3 State Variable Notation

It is convention to use $|$ to separate instructions and memory

$$\begin{aligned}\mathcal{P} &:= \{\{s_1, s_2, \dots, s_N\}, \{m_1, m_2, \dots, m_M\}\} \\ &= \{s_1, s_2, \dots, s_N | m_1, m_2, \dots, m_M\} \\ &= \{ \mathcal{I} | \mathcal{M} \}\end{aligned}$$

5 Boolean Methods

5.1 Definition of Boolean Methods

Define a boolean method; a method with inputs x_i and boolean output y

$$X_n = \{x_1, \dots, x_n\} : x_i \in \Omega \ \forall x_i \in X_n$$

$$P_{boolean}[X_n] := \{s_1, s_2, \dots, s_N \mid y, b_2, \dots, b_M\} \rightarrow y : y \in \{\mathbb{T}, \mathbb{F}\} \ \forall X_n \in \mathbb{D}_P$$

5.2 Abstraction Notation

$$\text{"Boolean Method"} := P_{boolean}[X_n] :$$

$$(P_{boolean}[X_n] \equiv \text{method}) \wedge (\exists \mathbb{D}_P : X_n \in \mathbb{D}_P \Leftrightarrow P_{boolean}[X_n] \rightarrow y \equiv \text{bool})$$

5.3 Boolean Method Output

The output of a boolean method y is read as "the result"

$$P_{boolean}[X_n] \rightarrow y$$

"Boolean Method $P_{boolean}$ with input(s) X_n outputs result y "

6 True or False Questions

6.1 Definition of True or False Question

Define a True or False Question; a bounded boolean expression given input(s) X_n

Given

$$X_n = \{x_1, x_2, \dots, x_n\}$$

$$Q^{T/F} := Q[X_n] : Q[X_n] \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_n \in \mathbb{D}_Q$$

A True or False Question is sometimes read as a "decision question"

6.2 Abstraction Notation

Given

$$X_n = \{x_1, x_2, \dots, x_n\}$$

$$\text{"True or False Question"} := Q^{T/F}[X_n] :$$

$$\exists \mathbb{D}_Q : (X_n \in \mathbb{D}_Q \Leftrightarrow Q^{T/F}[X_n] \equiv \text{bool})$$

7 Approach to a True or False Question

?? Does $(\mathbb{D}_S == \mathbb{D}_Q)$ or merely contained/bounded???

7.1 Definition of an Approach to a True or False Question

Given input(s) X_n and True or False Question $Q^{T/F}$; Define an approach to True or False Question $Q^{T/F}$; a boolean method bounded by the domain of $Q^{T/F}$

Given

$$X_n = \{x_1, x_2, \dots, x_n\}$$

$$Q^{T/F}[X_n]$$

$$S_Q[X_n] := P_Q[X_n] = \{s_1, s_2, \dots, s_N \mid y, b_2, \dots, b_M\} :$$

$$(P_Q[X_n] \rightarrow y) \wedge (y \in \{\mathbb{T}, \mathbb{F}\}) \quad \forall X_n \in \mathbb{D}_Q$$

7.2 Abstraction Notation

Given

$$X_n = \{x_1, x_2, \dots, x_n\}$$

$$Q^{T/F}[X_n]$$

$$\text{"Approach to True or False Question } Q \text{"} := S_Q[X_n] :$$

$$(S_Q[X_n] \equiv \equiv \text{boolean method}) \wedge (\mathbb{D}_S == \mathbb{D}_Q)$$

Solution to a True or False Question

7.3 Definition of Solution to a True or False Question

Given input(s) X_n and True or False Question $Q^{T/F}$;
Define a solution of True or False Question $Q^{T/F}$; an approach to $Q^{T/F}$
asserting output y is the answer for all input(s) in the domain of Q

Given

$$X_n = \{x_1, x_2, \dots, x_n\}$$
$$Q = Q^{T/F}[X_n]$$

$$S_Q^+[X_n] := P_Q[X_n] = \{s_1, s_2, \dots, s_N \mid y, b_2, \dots, b_M\} :$$
$$(P_Q[X_n] \rightarrow y) \wedge (y == a) \quad \forall X_n \in \mathbb{D}_Q$$

7.4 Abstraction Notation

Given

$$X_n = \{x_1, x_2, \dots, x_n\}$$
$$Q = Q^{T/F}[X_n]$$

$$\text{"Solution to True or False Question } Q\text{"} := S_Q^+[X_n] :$$
$$(S_Q^+[X_n] \equiv \text{approach to a True or False Question } Q) \wedge$$
$$(S_Q^+[X_n].output == |Q^{T/F}[X_n]|)$$

Incomplete Solution to a True or False Question

7.5 Definition of Incomplete Solution to a True or False Question

7.6 Abstraction Notation

8 Complexity of a Method

8.1 Definition of a Method's Complexity

Given

$$\mathcal{P} = \{s_1, s_2, \dots, s_N \mid y, b_2, \dots, b_M\}$$

$$\begin{aligned} \mathbf{O}_{\mathcal{P}} &:= \langle |\{s_1, s_2, \dots, s_N\}|, |\{y, b_2, \dots, b_M\}| \rangle \\ &= \langle |\mathcal{I}|, |\mathcal{M}| \rangle = \langle N, M \rangle \end{aligned}$$

8.2 Abstraction Notation

$$\text{"Method"} := \mathcal{P} :$$

$$(\mathcal{P} \equiv \text{Tensor}) \wedge (\mathcal{P} \supseteq \mathcal{I}) \wedge (\mathcal{P} \supseteq \mathcal{M})$$

1. By definition

$$\mathcal{P} \equiv \text{tensor}$$

2. By inheritance of Tensor

$$\mathcal{P} \underline{\supseteq} \text{size}$$

$\underline{\supseteq}$ is read as "has the attribute of"

3. Therefore size of \mathcal{P} is defined

$$\therefore |\mathcal{P}| := \langle |S_1|, |S_2|, \dots \rangle$$

4. By definition

$$(\mathcal{P} \supseteq \mathcal{I}) \wedge (\mathcal{P} \supseteq \mathcal{M})$$

5. Let

$$|\mathcal{P}| = \mathbf{O}_{\mathcal{P}} = \text{"Complexity of Method } \mathcal{P}\text{"}$$

6. By 3, 4, and 5

"Complexity of Method \mathcal{P} " = $\mathbf{O}_P := \langle |\mathcal{I}|, |\mathcal{M}| \rangle$

7. By definition of \mathcal{I}, \mathcal{M}

$$\mathcal{I} \equiv \text{Set} \wedge \mathcal{M} \equiv \text{Set}$$

8. By inheritance of Sets

$$\mathcal{I} \sqsupseteq \text{count} \wedge \mathcal{M} \sqsupseteq \text{count}$$

9. Therefore count of \mathcal{I} and \mathcal{M} is defined; by definition of \mathcal{I} and \mathcal{M}

$$\therefore |\mathcal{I}| = |\{s_1, s_2, \dots, s_N\}| := N; \quad |\mathcal{M}| = |\{y, b_2, \dots, b_M\}| := M$$

10. By 6 and 9

$$\text{"Complexity of Method } \mathcal{P}\text{"} = \mathbf{O}_P := \langle |\mathcal{I}|, |\mathcal{M}| \rangle = \langle N, M \rangle$$

9 Total Complexity of a Method

10 Simple Computational Complexity

The remainder of this document assumes simple computational complexity of dimension 2

10.1 Definition

Define simple computational complexity of dimension 2

$$\mathbf{O}[n] := \langle O_T[n], O_S[n] \rangle$$

10.2 Total Complexity

Define Total Complexity of solution s^+

$$\begin{aligned} O[n] &:= |s^+[X_n]| = |\{\mathcal{L}, \mathcal{M}, y_o\}| \\ &= |\mathcal{L}| + |\mathcal{M}| + |y_o| = N + M + 1 \end{aligned}$$

10.3 Time Complexity

Restate definition of Time Complexity $O_T[n]$ of solution s^+

$$\begin{aligned} s^+ &= \{\mathcal{L}, \mathcal{M}, y_o\} \\ O_T[n] &:= |\mathcal{L}| = N \end{aligned}$$

10.4 Space Complexity

Restate definition of Time Complexity $O_S[n]$ of solution s^+

$$\begin{aligned} s^+ &= \{\mathcal{L}, \mathcal{M}, y_o\} \\ O_S[n] &:= |\mathcal{M}| + |y_o| = M + 1 \end{aligned}$$

10.5 Total Complexity as a Function of Time and Space Complexity

$$\begin{aligned} O[n] &:= |s^+[X_n]| = |\{\mathcal{L}, \mathcal{M}, y_o\}| \\ &= |\mathcal{L}| + |\mathcal{M}| + |y_o| \\ &= O_T[n] + O_S[n] \end{aligned}$$

10.6 $O_S[n] > 0^*$

Assuming Program is not void

10.6.1 Proof

Assume $O_S[n] = 0$

$$O_S[n] := |\mathcal{M}| + |y_o|$$

$$O_S[n] = 0 \Rightarrow \mathcal{M} = y_o = \emptyset$$

$$y_o = \emptyset; \ y_o \in \{\mathbb{T}, \mathbb{F}\} \text{ by definition of } s^+$$

$\therefore O_S[n] = 0$ contradicts the definition of solution s^+ of a decision problem

$$O_S[n] \geq 0 \text{ by definition of magnitude}$$

$$\therefore O_S[n] > 0$$

10.7 $O_T[n] > 0^*$

Assuming Program is not void

10.7.1 Proof

Assume $O_T[n] = 0$

$$O_T[n] := |\mathcal{L}|$$

$$O_T[n] = 0 \Rightarrow y_o \notin \{\mathbb{T}, \mathbb{F}\}$$

$$y_o \notin \{\mathbb{T}, \mathbb{F}\}; \ y_o \in \{\mathbb{T}, \mathbb{F}\} \text{ by definition of } s^+$$

$\therefore O_T[n] = 0$ contradicts the definition of solution s^+ of a decision problem

$$O_T[n] \geq 0 \text{ by definition of magnitude}$$

$$\therefore O_T[n] > 0$$

10.8 $O[n] > 0^*$

Assuming Program is not void

10.8.1 Proof

$$O[n] := O_T[n] + O_S[n]$$

$$O_T[n] > 0; \quad O_S[n] > 0$$

$$\therefore O[n] > 0$$

10.9 $O[n] > O_T[n]^*$

Assuming Program is not void

10.9.1 Proof

$$O[n] := O_T[n] + O_S[n]$$

$$O_S[n] > 0$$

$$\therefore O[n] > O_T[n]$$

10.10 $O[n] > O_S[n]^*$

Assuming Program is not void

10.10.1 Proof

$$O[n] := O_T[n] + O_S[n]$$

$$O_T[n] > 0$$

$$\therefore O[n] > O_S[n]$$

10.11 $O[n+1] \geq O[n]$ **10.11.1 Proof**

$$X_i = \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\}$$

$$O[n] = |s^+[X_i]|$$

$$O[n+1] = \hat{O}[n] = |s^+[\hat{X}_i]|$$

For general solutions s^+

$$\begin{aligned}
s^+[\hat{X}_i] &\supseteq s^+[X_i] \\
\Rightarrow |s^+[\hat{X}_i]| &\geq |s^+[X_i]| \\
\therefore \hat{O}[n] = O[n+1] &\geq O[n]
\end{aligned}$$

11 Definition of Problem

Define Problem (also denoted as Question); a function with input(s) X_n and "answer" Y

$$X_n = \{x_1, \dots, x_n\}$$

11.1 Set of Questions

Define \mathbb{Q} ; the set of questions

$$\begin{aligned}\mathbb{Q} &:= \{Q_1, Q_2, \dots\} : \\ Q_i &= f[X_j] = Y_o \subseteq \Omega \quad \forall X_j, i\end{aligned}$$

11.2 Decision Questions / Decision Problems

11.2.1 Definition

Define decision problem; a function with inputs x_i and boolean output "answer" a_o

$$\begin{aligned}X_i &= \{x_1, \dots, x_n\} \\ D &:= f[X_i] = a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i\end{aligned}$$

11.3 Numerical Questions / Numerical Problems

11.3.1 Definition

Define numerical problem; a function with inputs x_i and numerical output y_o

$$\begin{aligned}X_i &= \{x_1, \dots, x_n\} \\ Q &:= f[X_i] = y_o \in \mathbb{R} \quad \forall X_i\end{aligned}$$

11.4 System Questions / System Problems

11.4.1 Definition

Define system problem; a function with inputs x_i and outputs y_j

$$\begin{aligned}X_i &= \{x_1, \dots, x_n\} \\ Q &:= f[X_i] = Y_o = \{y_1, \dots, y_m\} \quad \forall X_i\end{aligned}$$

12 Solutions

12.1 Definition

Program P is a solution s^+ to decision problem D if

1. P outputs answer a_o for all inputs $X_i \forall i$
and
2. $s^+[X_i]$ is a subset of $s^+[\hat{X}_i]$

$$\begin{aligned}
 X_i &= \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_n, x_{n+1}, C\} \\
 D &:= f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i \\
 s^+ &= s^+[X_i] := P : \\
 (P[X_i] \rightarrow y_o == a_o \quad \forall X_i) &\cap (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i) \\
 P[X_i] &= \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} \\
 s^+ = P[X_i] &= \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} \quad \forall X_i
 \end{aligned}$$

12.1.1 Property of No-op ;

No-op ; can be added to any solution S_i without modifying the output y_o or memory b_i

$$\begin{aligned}
 & ::= \emptyset \\
 s^+ &= \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} \\
 \hat{s}^+ &= \{s_1, s_2, \dots, ;, \dots, s_{O_T[n]+1}, \hat{b}_1, \hat{b}_2, \dots, \hat{b}_{O_S[n]}, \hat{y}_o\} \\
 \hat{y}_o &= y_o \quad \forall k
 \end{aligned}$$

12.2 Definition of S^+

Define S^+ ; the set of solutions to decision problem D

$$\begin{aligned}
 X_i &= \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_n, x_{n+1}, C\} \\
 D &:= f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i \\
 s_j^+ &= s_j^+[X_i] := P : \\
 (P[X_i] \rightarrow y_o == a_o \quad \forall X_i) &\cap (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i) \\
 S^+ &:= \{s_j^+, \dots\} \quad \forall j
 \end{aligned}$$

12.3 Definition of Solvable

Define solvable

$$\begin{aligned}
X_i &= \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_n, x_{n+1}, C\} \\
D &:= f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i \\
\text{solvable} &= \text{solvable}[D] = b_o \in \{\mathbb{T}, \mathbb{F}\} := \\
\exists P : (P[X_i] \rightarrow y_o == a_o \quad \forall X_i) &\cap (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)
\end{aligned}$$

13 The set of all Decision Problems \mathbb{D}

13.1 Definition

Define the set of decision problems \mathbb{D}

$$\begin{aligned}
X_i &= \{x_1, \dots, x_n, C\} \\
D_j &:= f_j[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i \\
\mathbb{D} &:= \{D_j, \dots\} \quad \forall j
\end{aligned}$$

14 Complexity

14.1 Time Complexity of a Decision Problem $O_T[n]$

Define Time Complexity $O_T[n]$ of solution s^+ to Decision Problem D as the total number of logical operations

$$\begin{aligned}
X_i &= \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\} \\
D &:= f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i \\
s^+[X_i] &:= P : \\
(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) &\cap (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i) \\
s^+ &= \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} \\
&= \{\mathcal{L}, \mathcal{M}, y_o\} \\
O_T[n] &:= |\mathcal{L}| = N
\end{aligned}$$

14.2 Space Complexity $O_S[n]$

Define Space Complexity $O_S[n]$ of solution s^+ to Decision Problem D as the total number of memory elements

$$X_i = \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\}$$

$$D := f[X_i] \rightarrow a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$

$$s^+[X_i] := P :$$

$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \cap (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

$$\begin{aligned} s^+ &= \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} \\ &= \{\mathcal{L}, \mathcal{M}, y_o\} \end{aligned}$$

$$O_S[n] := |\mathcal{M}| + |y_o|^* = M + 1$$

*It is convention to reserve one memory element for output y_o .
Void programs do not require the y_o memory element for output

15 Definition of Complexity

Define Complexity $O[n]$ as a vector of dimension V

$$\mathbf{O}[n] := \langle O_T[n], O_S[n], O_3[n], O_4[n], \dots, O_V[n] \rangle$$

16 Complexity of Canonical Instructions

$$c := a \leftarrow l[X_n]$$

17 Complexity of Computational Operations

17.1 +

Express the bounds of complexity for Computational Operation +

18 Inductive Functions

18.1 Inductive Function f_{n+1}

$$\begin{aligned} O[n] &= O_T[n] + O_S[n] \\ O[n+1] &= O_T[n+1] + O_S[n+1] \\ f_{n+1}[n] &:= O[n+1] - O[n] \end{aligned}$$

18.2 Inductive Space and Time Formulas

$$\begin{aligned} f_{n+1}^T[n] &:= O_T[n+1] - O_T[n] \\ O_T[n+1] &= O_T[n] + f_{n+1}^T[n] \\ f_{n+1}^S[n] &:= O_S[n+1] - O_S[n] \\ O_S[n+1] &= O_S[n] + f_{n+1}^S[n] \end{aligned}$$

18.3 Inductive Function Expressions

Relate $f_{n+1}[n]$ to equivalence functions

$$\begin{aligned} O[n] &= O_T[n] + O_S[n] \\ O[n+1] &= O_T[n+1] + O_S[n+1] = O[n] + f_{n+1}[n] \\ O_T[n] &= O[n] - O_S[n] \\ O_S[n] &= O[n] - O_T[n] \end{aligned}$$

$$\begin{aligned} f_{n+1}[n] &= O[n+1] - O[n] \\ f_{n+1}[n] &= O_T[n+1] + O_S[n+1] - O[n] \\ f_{n+1}[n] &= O_T[n+1] - O_T[n] + O_S[n+1] - O_S[n] \\ f_{n+1}[n] &= O[n+1] - O_T[n] - O_S[n] \\ f_{n+1}[n] &= f_{n+1}^T[n] + f_{n+1}^S[n] \end{aligned}$$

18.4 Zero Order Space Inductive Function

$$\text{Let } O_S[n] \sim n^0$$

$$f_{n+1}[n] = O_T[n+1] - O_T[n] + O_S[n+1] - O_S[n] = O_T[n+1] - O_T[n]$$

19 Polynomial Complexity

19.1 Definition

Decision problem D with solution s^+ has polynomial total complexity $O[n]$ if

$$\begin{aligned} & \exists K, C, \lambda_1 \dots \lambda_K : \\ & O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \end{aligned}$$

19.2 Polynomial Problems

Define \mathbb{P} , the set of Decision Problems that can be solved with Polynomial Complexity

$$\begin{aligned} & \mathbb{P} := \{D_1, D_2, \dots\} : \\ & \exists K, C, \lambda_1 \dots \lambda_K : \\ & O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n, D_i \in \mathbb{P} \end{aligned}$$

19.3 Polynomial Order of Complexity

Solution s^+ with total complexity $O[n]$ is said to be of order n^K

$$\begin{aligned} & O[n] \sim n^K \\ & O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \end{aligned}$$

19.4 Property of Polynomial Complexity 1

Solutions with polynomial complexity have convergent complexity

$$\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} = 1$$

19.4.1 Proof

$$\begin{aligned} & O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \\ & O[n+1] = (\lambda_K (n+1))^K + (\lambda_{K-1} (n+1))^{K-1} \dots + \lambda_1 (n+1) + C \\ & = (\lambda_K n)^K + (\tilde{\lambda}_{K-1} n)^{K-1} \dots + \tilde{\lambda}_1 n + \tilde{C} \\ & \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} \\ & = \lim_{n \rightarrow \infty} \frac{(\lambda_K n)^K + (\tilde{\lambda}_{K-1} n)^{K-1} \dots + \tilde{\lambda}_1 n + \tilde{C}}{(\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C} \end{aligned}$$

$$\begin{aligned}
&= \lim_{n \rightarrow \infty} \frac{(\lambda_K n)^K}{(\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C} + \frac{(\tilde{\lambda}_{K-1} n)^{K-1}}{(\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C} + \dots + \\
&\quad \frac{\tilde{\lambda}_1 n}{(\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C} + \frac{\tilde{C}}{(\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C} \\
&= 1 = \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]}
\end{aligned}$$

19.5 Property of Polynomial Complexity 2

$$\begin{aligned}
&\exists K, \hat{C}, \hat{\lambda}_1, \dots, \hat{\lambda}_{K-1} : \\
O[n+1] - O[n] &= f_{n+1}[n] = (\hat{\lambda}_{K-1} n)^{K-1} \dots + \hat{\lambda}_1 n + \hat{C} \quad \forall n
\end{aligned}$$

19.5.1 Proof

$$\begin{aligned}
O[n] &= (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \\
O[n+1] &= (\lambda_K (n+1))^K + (\lambda_{K-1} (n+1))^{K-1} \dots + \lambda_1 (n+1) + C \\
&= (\lambda_K n)^K + (\tilde{\lambda}_{K-1} n)^{K-1} \dots + \tilde{\lambda}_1 n + \tilde{C} \\
O[n+1] - O[n] &= ((\tilde{\lambda}_{K-1} - \lambda_{K-1}) n)^{K-1} \dots + (\tilde{\lambda}_1 - \lambda_1) n + (\tilde{C} - C) \\
O[n+1] - O[n] &= (\hat{\lambda}_{K-1} n)^{K-1} \dots + \hat{\lambda}_1 n + \hat{C}
\end{aligned}$$

19.6 Property of Polynomial Complexity 3

$$\lim_{n \rightarrow \infty} \frac{f_{n+1}[n]}{O[n]} = 0$$

19.6.1 Proof

$$\begin{aligned}
\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} &= 1 \\
\lim_{n \rightarrow \infty} \frac{O[n] + f_{n+1}[n]}{O[n]} &= 1 \\
\lim_{n \rightarrow \infty} \frac{O[n]}{O[n]} + \frac{f_{n+1}[n]}{O[n]} &= 1 \\
\lim_{n \rightarrow \infty} 1 + \frac{f_{n+1}[n]}{O[n]} &= 1 \\
\lim_{n \rightarrow \infty} \frac{f_{n+1}[n]}{O[n]} &= 0
\end{aligned}$$

19.7 Total Polynomial Complexity Implies Time bounded Polynomial Complexity

$$D \in \mathbb{P} \implies O_T[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C$$

19.7.1 Proof

$$\begin{aligned}
O[n] &= (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \\
O[n] &:= O_T[n] + O_S[n]; \quad O_S[n] > 0 \\
\therefore O_T[n] &< (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n
\end{aligned}$$

19.8 Total Polynomial Complexity Implies Space bounded Polynomial Complexity

$$D \in \mathbb{P} \implies O_S[n] < (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C$$

19.8.1 Proof

$$\begin{aligned}
O[n] &= (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \\
O[n] &:= O_T[n] + O_S[n]; \quad O_T[n] > 0 \\
\therefore O_S[n] &< (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n
\end{aligned}$$

19.9 Polynomial Complexity in Space and Time Implies Polynomial Total Complexity

$$\begin{aligned}
(O_S[n] &= (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + \lambda_0) \\
&\quad \wedge \\
(O_T[n] &= (\hat{\lambda}_M n)^M + (\hat{\lambda}_{M-1} n)^{M-1} \dots + \hat{\lambda}_1 n + \hat{\lambda}_0) \\
&\implies D \in \mathbb{P}
\end{aligned}$$

19.9.1 Proof

$$\begin{aligned}
O_S[n] &= \lambda_K n^K + \lambda_{K-1} n^{K-1} + \dots + \lambda_1 n + \lambda_0 \\
O_T[n] &= \hat{\lambda}_M n^M + \hat{\lambda}_{M-1} n^{M-1} + \dots + \hat{\lambda}_1 n + \hat{\lambda}_0 \\
O[n] &= O_S[n] + O_T[n] \\
*O[n] &= (\hat{\lambda}_0 + \lambda_0) + n(\lambda_1 + \hat{\lambda}_1) + \dots + n^K(\lambda_K + \hat{\lambda}_K) + \hat{\lambda}_{K+1} n^{K+1} + \dots + \hat{\lambda}_M n^M \\
\therefore O[n] &\text{ has polynomial total complexity by definition}
\end{aligned}$$

* Assume $K < M$, similar proof for $K=M$, $K>M$

20 Non-Polynomial Complexity

20.1 Definition

Decision problem \tilde{D} with solution s^+ has non-polynomial total complexity $O[n]$ if

$$\begin{aligned} \nexists K, \lambda_0, \lambda_1, \dots, \lambda_K : \\ O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \end{aligned}$$

20.2 Non-Polynomial Problems

Define \mathcal{N} , the set of Decision Problems that cannot be solved with Polynomial Complexity

$$\begin{aligned} \mathcal{N} := \{\tilde{D}_1, \tilde{D}_2, \dots\} : \\ \nexists K, C, \lambda_1 \dots \lambda_K : \\ O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n, s^+ \in S_i^+, \tilde{D}_i \in \mathcal{N} \end{aligned}$$

20.3 \mathbb{P} and \mathcal{N} are disjoint

$$\mathbb{P} \cap \mathcal{N} = \emptyset$$

20.3.1 Proof

Let $D \in \mathcal{N}$

$$\begin{aligned} \nexists K, C, \lambda_1 \dots \lambda_K : \\ O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \end{aligned}$$

Assume $D \in \mathbb{P}$

$$\begin{aligned} \exists K, C, \lambda_1 \dots \lambda_K : \\ O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \end{aligned}$$

Contradicts the definition of \mathcal{N}

$$\therefore D \in \mathcal{N} \Rightarrow D \notin \mathbb{P}$$

Let $D \in \mathbb{P}$

$$\begin{aligned} & \exists K, C, \lambda_1 \dots \lambda_K : \\ O[n] &= (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \end{aligned}$$

Assume $D \in \mathcal{N}$

$$\begin{aligned} & \nexists K, C, \lambda_1 \dots \lambda_K : \\ O[n] &= (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n \end{aligned}$$

Contradicts the definition of \mathbb{P}

$$\therefore D \in \mathbb{P} \Rightarrow D \notin \mathcal{N}$$

$$\begin{aligned} D \in \mathcal{N} &\Rightarrow D \notin \mathbb{P}; D \in \mathbb{P} \Rightarrow D \notin \mathcal{N} \\ \therefore \mathbb{P} \cap \mathcal{N} &= \emptyset \end{aligned}$$

21 Discrete Derivative; Z Transform

21.1 Discrete Derivative

Define derivative for discrete function $f[n]$

$$\Delta_n^1 f[n] := f[n+1] - f[n]$$

We will use the above definition for the remainder of this document

21.2 Zero Order Derivative

$$\Delta_n^0 f[n] = f[n]$$

21.3 K^{th} Discrete Derivative

Define the K^{th} derivative of discrete function $f[n]$

$$\Delta_n^K f[n] := \Delta_n^{K-1} f[n+1] - \Delta_n^{K-1} f[n]$$

21.4 K^{th} Discrete Derivative as an Alternating Sum

$$\begin{aligned}
\Delta_n^K f[n] &:= \Delta_n^{K-1} f[n+1] - \Delta_n^{K-1} f[n] \\
&= (\Delta_n^{K-2} f[n+2] - \Delta_n^{K-2} f[n+1]) - (\Delta_n^{K-2} f[n+1] - \Delta_n^{K-2} f[n]) \\
&= (\Delta_n^{K-2} f[n+2] - 2\Delta_n^{K-2} f[n+1] + \Delta_n^{K-2} f[n]) \\
&= \sum_{i=0}^K (-1)^i \binom{K}{i} \Delta_n^0 f[n+i] \\
&= \sum_{i=0}^K (-1)^i \binom{K}{i} f[n+i]
\end{aligned}$$

21.5 Z Transform

Define the Z Transform for discrete function $f[n]$

$$\mathcal{Z}(f[n]) := \sum_{n=0}^{\infty} f[n] z^{-n}$$

21.6 Z Transform of 0 Order Derivative

$$\begin{aligned}
\Delta_n^0 f[n] &:= f[n] \\
\mathcal{Z}(\Delta_n^0 f[n]) &= \mathcal{Z}(f[n])
\end{aligned}$$

21.7 Z Transform of 1st Derivative

$$\begin{aligned}
\Delta_n^1 f[n] &:= f[n+1] - f[n] \\
\mathcal{Z}(\Delta_n^1 f[n]) &= \mathcal{Z}(f[n+1] - f[n]) \\
&= \sum_{n=0}^{\infty} (f[n+1] - f[n]) z^{-n} \\
&= \sum_{n=0}^{\infty} (f[n+1] z^{-n} - f[n] z^{-n}) \\
&= \sum_{n=0}^{\infty} f[n+1] z^{-n} - \sum_{n=0}^{\infty} f[n] z^{-n} \\
&= \sum_{m=0}^{\infty} f[m+1] z^{-m} - \sum_{n=0}^{\infty} f[n] z^{-n}
\end{aligned}$$

Let

$$\begin{aligned}
\hat{m} &= m+1; \quad m = \hat{m}-1 \\
&= \sum_{m=0}^{\infty} f[\hat{m}] z^{-(\hat{m}-1)} - \mathcal{Z}(f[n]) \\
&= z^1 \sum_{\hat{m}=1}^{\infty} f[\hat{m}] z^{-\hat{m}} - \mathcal{Z}(f[n])
\end{aligned}$$

$$\begin{aligned}
&= z^1 \sum_{\hat{m}=1}^{\infty} f[\hat{m}] z^{-\hat{m}} + f[0] - f[0] - \mathcal{Z}(f[n]) \\
&= z^1 \sum_{\hat{m}=0}^{\infty} f[\hat{m}] z^{-\hat{m}} - f[0] - \mathcal{Z}(f[n]) \\
&= z^1 \mathcal{Z}(f[n]) - f[0] - \mathcal{Z}(f[n]) \\
&\mathcal{Z}(\Delta_n^1 f[n]) = \mathcal{Z}(f[n])(z^1 - 1) - f[0]
\end{aligned}$$

21.8 Z Transform of K^{th} Derivative

$$\begin{aligned}
\mathcal{Z}(f[n]) &:= \sum_{n=0}^{\infty} f[n] z^{-n} \\
\mathcal{Z}(\Delta_n^K f[n]) &= \sum_{n=0}^{\infty} \Delta_n^K f[n] z^{-n} \\
&= \sum_{n=0}^{\infty} \sum_{i=0}^K (-1)^i ({}_K C_i) f[n+i] z^{-n} \\
&= \sum_{n=0}^{\infty} (f[n+K] - ({}_K C_1) f[n+K-1] + ({}_K C_2) f[n+K-2] - \dots \pm f[n]) z^{-n} \\
&= \sum_{n=0}^{\infty} f[n+K] z^{-n} - ({}_K C_1) f[n+K-1] z^{-n} + ({}_K C_2) f[n+K-2] z^{-n} - \dots \pm f[n] z^{-n} \\
&= z^K \mathcal{Z}(f[n]) + \sum_{i=0}^{K-1} f[i] - ({}_K C_1) z^{K-1} \mathcal{Z}(f[n]) - \sum_{j=0}^{K-2} f[j] + \\
&\quad ({}_K C_2) z^{K-2} \mathcal{Z}(f[n]) + \sum_{k=0}^{K-3} f[k] - \dots \pm \mathcal{Z}(f[n])
\end{aligned}$$

When K is odd

$$\mathcal{Z}(\Delta_n^K f[n]) = (z-1)^K \mathcal{Z}(f[n]) + \sum_{i=0}^{\frac{n+1}{2}} f[2i] \quad K > 0$$

When K is even

$$\mathcal{Z}(\Delta_n^K f[n]) = (z-1)^K \mathcal{Z}(f[n]) + \sum_{j=0}^{\frac{n}{2}} f[2j+1] \quad K > 0$$

22 Divergent Complexity

22.1 Definition of Converges to

??Can $f[n] = 0$??

$$f[n] \text{ converges to } C = \text{convergent}[f[n], C] = a_o; a_o \in \{\mathbb{T}, \mathbb{F}\} =$$

$$|C - f[n+1]| < |C - f[n]| \quad \forall n$$

$$\begin{aligned} & \wedge \\ \nexists K : |C - f[\hat{n}]| > K \quad \forall n; K > 0 \end{aligned}$$

22.1.1 Notation

C is commonly denoted by a limit

$$C = \lim_{n \rightarrow \infty} f[n]$$

22.2 Definition of General Convergence

$$f[n] \text{ is } \textit{convergent} = \textit{convergent}[f[n]] = a_o; a_o \in \{\mathbb{T}, \mathbb{F}\} =$$

$$\begin{aligned} & \exists C : \\ & \textit{convergent}[f[n], C] == \mathbb{T} \end{aligned}$$

Alternatively

$$\begin{aligned} f[n] \text{ is } \textit{convergent} &= \textit{convergent}[f[n]] = a_o \in \{\mathbb{T}, \mathbb{F}\} = \\ & \exists C : \\ & f[n] \textit{ converges to } C \end{aligned}$$

22.3 Definition of Divergence

$$\begin{aligned} \textit{diverges}[f[n]] &= \neg \textit{converges}[f[n]] = d_o; d_o \in \{\mathbb{T}, \mathbb{F}\} \\ &:= \nexists C : \textit{convergent}[f[n], C] == \mathbb{T} \end{aligned}$$

22.4 Alternate Definition of Divergence

$$\begin{aligned} \textit{diverges}[f[n]] &= \neg \textit{converges}[f[n]] = d_o; d_o \in \{\mathbb{T}, \mathbb{F}\} \\ &= \textit{convergent}[f[n], C] == \mathbb{F} \quad \forall C \end{aligned}$$

22.4.1 Proof of Equivalence; Alternate Definition of Divergence

22.5 ?Necessary or Sufficient? Criteria 1 For Divergence

? The derivative as a function of K ? Function f[n] diverges if the K^{th} derivative of f[n] is strictly increasing

$$\text{diverges}[f[n]] := \nexists C : \text{convergent}[f[n], C] == \mathbb{T}$$

$$\Longleftrightarrow$$

$$\Delta_n^{K+1} f[n] > \Delta_n^K f[n] \quad \forall K$$

Alternatively

$$\Delta_n^{K+1} f[n] - \Delta_n^K f[n] > 0 \quad \forall K$$

$$\Delta_n^{K+2} > 0 \quad \forall K$$

22.5.1 Criteria 1; Proof of Necessity and Sufficiency

$$\text{diverges}[f[n]] = d_o; d_o \in \{\mathbb{T}, \mathbb{F}\}$$

$$= \nexists C : \text{convergent}[f[n], C] == \mathbb{T}$$

Let

$$f[n] :$$

$$\Delta_n^{K+2} > 0 \quad \forall K$$

22.6 ?Necessary or Sufficient? Criteria 2 For Divergence

Function $f[n]$ diverges if the Derivative as a function of K does not Converge

$$f[n] \text{ is Divergent} = \text{Divergent}[f[n]] = a_o \in \{\mathbb{T}, \mathbb{F}\} :=$$

$$\nexists c : \lim_{n \rightarrow \infty} \Delta_n^K f[n] = c$$

22.6.1 Criteria 2; Proof of Necessity and Sufficiency

$$\text{diverges}[f[n]] = d_o; d_o \in \{\mathbb{T}, \mathbb{F}\}$$

$$= \nexists C : \text{convergent}[f[n], C] == \mathbb{T}$$

22.7 Verbal Expressions

$$f[n] \text{ diverges} = f[n] \text{ is divergent} =$$

$$f[n] \text{ is not convergent} = f[n] \text{ does not converge}$$

22.8 Definition of Divergent Function

22.8.1 Definition 1

Define Divergent Function $f[n]$ having strictly increasing K^{th} derivative

$$f[n] \text{ is Divergent} = Divergent[f[n]] = a_o \in \{\mathbb{T}, \mathbb{F}\} :=$$

$$\Delta_n^{K+1} f[n] > \Delta_n^K f[n] \quad \forall K$$

Alternatively

$$\Delta_n^{K+1} f[n] - \Delta_n^K f[n] > 0 \quad \forall K$$

$$\Delta_n^{K+2} > 0 \quad \forall K$$

22.8.2 Definition 2

$$f[n] \text{ is Divergent} = Divergent[f[n]] = a_o \in \{\mathbb{T}, \mathbb{F}\} :=$$

$$\nexists c : \lim_{n \rightarrow \infty} \Delta_n^K f[n] = c$$

22.8.3 Proof of Equivalence Definition 1 \Leftrightarrow Definition 2

22.8.4 Sufficient Proof

$$f[n] \text{ is Divergent} \Leftrightarrow$$

$$\nexists c : \lim_{n \rightarrow \infty} \Delta_n^K f[n] = c$$

22.8.5 Necessary Proof

$$f[n] \text{ is Divergent} \Leftrightarrow$$

$$\nexists c : \lim_{n \rightarrow \infty} \Delta_n^K f[n] = c$$

Proof by contradiction of definition of limit Using the definition of increasing convergence for a discrete function*

22.9 Defintion

Decision problem \hat{D} with solution s^+ has divergent total complexity $O[n]$ if

$$\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} \text{ diverges}$$

22.10 Divergent Problems

$$\begin{aligned}\mathcal{D} &:= \{\hat{D}_1, \hat{D}_2, \dots\} : \\ \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} &\text{ diverges } \forall s^+ \in S_i^+, \hat{D}_i \in \hat{\mathcal{D}}\end{aligned}$$

22.11 The Set of Polynomial Solutions and the Set of Divergent Solutions are disjoint

$$\mathbb{P} \cap \hat{\mathcal{D}} = \emptyset$$

22.12 Proof

Let $D \in \hat{\mathcal{D}}$

$$\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} \text{ diverges by definition}$$

Assume $D \in \mathbb{P}$

$$\begin{aligned}\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} &= 1 \\ \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} &= 1 \text{ contradicts the definition of Divergent Problems} \\ \therefore D \in \hat{\mathcal{D}} &\Rightarrow D \notin \mathbb{P}\end{aligned}$$

Let $D \in \mathbb{P}$

$$\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} = 1 \text{ by property of Polynomial complexity}$$

Assume $D \in \hat{\mathcal{D}}$

$$\begin{aligned}\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} &\text{ diverges} \\ \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} &\text{ diverges contradicts a property of Polynomial complexity} \\ \therefore D \in \mathbb{P} &\Rightarrow D \notin \hat{\mathcal{D}} \\ \therefore \mathbb{P} \cap \hat{\mathcal{D}} &= \emptyset\end{aligned}$$

23 Subprograms

23.1 Definition of Subprogram

Define a Subprogram of Program P; a subset of Program P

$$P := \{s_1, s_2, \dots, s_N, b_1, b_2, \dots, b_M, y_o\} = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$P_{sub} := \tilde{P} \mid$$

$$\tilde{P} \subseteq P$$

23.2 Identity Subprogram

23.2.1 Definition

23.2.2 Prove the Identity Subprogram is a Subprogram of P

23.3 Restate the subprogram condition of general solutions

Recall the definition of general solution s^+

$$X_i = \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\}$$

$$s^+ = s^+[X_i] := P :$$

$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

The subprogram condition is one of two conditions for a general solution

$$P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i$$

The term subprogram is used interchangeably with the term subfunction

23.4 Prove $O[n]$ is a non-decreasing function

Consider solution s^+ with complexity $O[n]$

$$X_i = \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\}$$

$$s^+ = s^+[X_i] := P :$$

$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

$$s^+ = \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\}$$

$$= \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$\begin{aligned}
O[n] &:= O_T[n] + O_S[n] \\
O_T[n] &:= |\mathcal{L}| = N \\
O_S[n] &:= |\mathcal{M}| + |y_o| = M + 1
\end{aligned}$$

$O[n+1]$ denotes the total complexity for solution $s^+[\hat{X}_i]$

$$s^+[\hat{X}_i] = \hat{s}^+$$

Let

$$\begin{aligned}
O[n+1] &< O[n] \\
&\Rightarrow \hat{N} + \hat{M} < N + M \\
\hat{s}^+ &= \{s_1, s_2, \dots, s_{\hat{N}} | b_1, b_2, \dots, b_{\hat{M}}, y_o\} \\
&\Rightarrow \hat{s}^+ \not\supseteq s^+ \\
P[\hat{X}_i] &\not\supseteq P[X_i] \quad \forall X_i, \hat{X}_i
\end{aligned}$$

$\therefore O[n+1] < O[n]$ contradicts the definition of solution s^+

$$O[n+1] \geq O[n]$$

23.5 Definition of Subfunction

$$\begin{aligned}
X_i &= \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\} \\
s^+ &= s^+[X_i] := P : \\
(P[X_i] \rightarrow y_o = a_o \quad \forall X_i) &\cap (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i) \\
s^+ &= \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} \\
&= \{\mathcal{L}, \mathcal{M}, y_o\}
\end{aligned}$$

$$\begin{aligned}
Sub[X_i] &:= S = \{s_j, \dots | b_k, \dots, y_o\} : \\
s_j, b_k &\in s^+ \quad \forall s_j, b_k \in S
\end{aligned}$$

23.5.1 $s^+[X_i]$ is a subfunction of $s^+[\hat{X}_i]$

$$s^+ = \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\}$$

$$\hat{s}^+ = \{s_1, s_2, \dots, s_N, \dots, s_{\hat{N}} | b_1, b_2, \dots, b_M, \dots, b_{\hat{M}}, y_o\}; \quad \hat{N} + \hat{M} \geq N + M$$

By definition of solution

$$\hat{s}^+ = P[\hat{X}_i] \supseteq P[X_i] = s^+ \quad \forall X_i, \hat{X}_i$$

$$\Rightarrow s_j, b_k \in \hat{s}^+ \quad \forall s_j, b_k \in s^+$$

23.6 Subfunction Decomposition of Solutions

FIX Double check conditions!!! Solutions s^+ can be written as the union of subfunctions $Sub_k[X_i]$

$$X_i = \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\}$$

$$s^+ = s^+[X_i] := P :$$

$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i)$$

$$s^+ = \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\}$$

$$= \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$s^+ = Sub_1[X_i] \cup Sub_2[X_i] \cup \dots \cup Sub_z[X_i]$$

$$= \{\mathcal{L}_1 | \mathcal{M}_1, y_o\} \cup \{\mathcal{L}_2 | \mathcal{M}_2, y_o\} \cup \dots \cup \{\mathcal{L}_z | \mathcal{M}_z, y_o\} :$$

$$\mathcal{L}_j \cap \mathcal{L}_k = \emptyset \quad \forall j, k \neq j$$

$$s^+ = \{s_1^1, \dots, s_{N_1}^1 | b_1^1, \dots, y_o\} \cup \{s_1^2, \dots, s_{N_2}^2 | b_1^2, \dots, y_o\} \cup \dots \cup \{s_1^z, \dots, s_{N_z}^z | b_1^z, \dots, y_o\} :$$

$$\sum_{l=1}^z N_l = N = O_T[n]$$

24 Subfunction Complexity

24.1 Disjoint Subfunction Operations

$$\mathcal{L}_i \cap \mathcal{L}_j = \emptyset \quad \forall i, j \neq i$$

24.2 Shared Subfunction Memory

$$|\mathcal{M}_i \cap \mathcal{M}_j| \geq 0 \quad \forall i, j \neq i$$

24.2.1 Time Complexity of Subfunctions

Subfunction time complexity is additive

$$s^+ = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$Sub_i[X] := S_i = \{s_j, \dots | b_k, \dots, y_o\} :$$

$$s_j, b_k \in s^+ \quad \forall s_j, b_k \in S_i$$

$$s^+ = \{\mathcal{L}_1 | \mathcal{M}_1, y_o\} \cup \{\mathcal{L}_2 | \mathcal{M}_2, y_o\} \cup \dots \cup \{\mathcal{L}_z | \mathcal{M}_z, y_o\} :$$

$$\mathcal{L}_i \cap \mathcal{L}_j = \emptyset \quad \forall i, j \neq i$$

$$\mathcal{L} = \cup_{i=1}^z \mathcal{L}_i$$

$$\mathcal{L}_i \cap \mathcal{L}_j = \emptyset \quad \forall i, j \neq i$$

$$O_T[n] = |\mathcal{L}| = N$$

$$O_T[n] = |\cup_{i=1}^z \mathcal{L}_i| = \sum_{i=1}^z |\mathcal{L}_i|^* = |\mathcal{L}_1| + |\mathcal{L}_2| + \dots + |\mathcal{L}_z|$$

$$= O_{T_1}[n] + O_{T_2}[n] + \dots + O_{T_z}[n] = N_1 + N_2 + \dots + N_z$$

*Due to the disjoint condition of subfunction operations $\mathcal{L}_i \cap \mathcal{L}_j = \emptyset \quad \forall i, j \neq i$

24.2.2 Space Complexity of Subfunctions

Subfunctions can access the full memory \mathcal{M} with no added space complexity

$$s^+ = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$Sub_i[X] := S_i = \{s_j, \dots | b_k, \dots, y_o\} :$$

$$s_j, b_k \in s^+ \quad \forall s_j, b_k \in S_i$$

$$s^+ = \{\mathcal{L}_1 | \mathcal{M}_1, y_o\} \cup \{\mathcal{L}_2 | \mathcal{M}_2, y_o\} \cup \dots \cup \{\mathcal{L}_z | \mathcal{M}_z, y_o\} :$$

$$\mathcal{L}_i \cap \mathcal{L}_j = \emptyset \quad \forall i, j \neq i$$

$$s^+ = \{\mathcal{L}_1 | \mathcal{M}, y_o\} \cup \{\mathcal{L}_2 | \mathcal{M}, y_o\} \cup \dots \cup \{\mathcal{L}_z | \mathcal{M}, y_o\} :$$

$$\mathcal{L}_i \cap \mathcal{L}_j = \emptyset \quad \forall i, j \neq i$$

$$\mathcal{M} = \cup_{i=1}^z \mathcal{M}_i = \cup_{i=1}^z \mathcal{M}$$

$$O_S[n] = |\mathcal{M}| = M$$

$$O_S[n] = |\cup_{i=1}^z \mathcal{M}_i| = M$$

24.2.3 Shared State Notation

$$s^+ = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$Sub_i[X] := S_i = \{s_j, \dots | b_k, \dots, y_o\} :$$

$$s_j, b_k \in s^+ \quad \forall s_j, b_k \in S_i$$

$$s^+ = \{\mathcal{L}_1 | \mathcal{M}, y_o\} \cup \{\mathcal{L}_2 | \mathcal{M}, y_o\} \cup \dots \cup \{\mathcal{L}_z | \mathcal{M}, y_o\} :$$

$$\mathcal{L}_i \cap \mathcal{L}_j = \emptyset \quad \forall i, j \neq i$$

25 Polynomial Solution Subfunction Properties

25.1 Restate Definition of Subfunction

$$X_n = \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\}$$

$$s^+ = s^+[X_n] := P :$$

$$(P[X_n] \rightarrow y_o == a_o \quad \forall X_i) \quad \cap \quad (P[\hat{X}_n] \supseteq P[X_n] \quad \forall X_n, \hat{X}_n)$$

$$\begin{aligned} s^+ &= \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} \\ &= \{\mathcal{L}, \mathcal{M}, y_o\} \end{aligned}$$

$$Sub[X_n] := S = \{s_j, \dots | b_k, \dots, y_o\} :$$

$$s_j, b_k \in s^+ \quad \forall s_j, b_k \in S$$

25.2 Property of Polynomial Solution Subfunctions

Let

$$D \in \mathbb{P}$$

$$X_n = \{x_1, \dots, x_n, C\}; \quad \hat{X}_n = \{x_1, \dots, x_{n+1}, C\}$$

$$s^+ = s^+[X_n] := P :$$

$$(P[X_i] \rightarrow y_o == a_o \quad \forall X_n) \quad \cap \quad (P[\hat{X}_n] \supseteq P[X_n] \quad \forall X_n, \hat{X}_n)$$

$$\exists K, C, \lambda_1 \dots \lambda_K \quad :$$

$$O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n$$

$$s^+ = Sub_1[X_n] \cup Sub_2[X_n] \cup \dots \cup Sub_z[X_n]$$

$$\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} = 1$$

$$= \lim_{n \rightarrow \infty} \frac{O_T^1[n+1] + O_T^2[n+1] + \dots + O_T^z[n+1] + O_S[n+1]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]}$$

$$\lim_{n \rightarrow \infty} \frac{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n] + f_{T_{n+1}}^1[n] + f_{T_{n+1}}^2[n+1] + \dots + f_{T_{n+1}}^z[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} =$$

$$\begin{aligned}
&= \lim_{n \rightarrow \infty} 1 + \frac{f_{T_{n+1}}^1[n] + f_{T_{n+1}}^2[n+1] + \dots + f_{T_{n+1}}^z[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} = 1 \\
&\Rightarrow \lim_{n \rightarrow \infty} \frac{f_{T_{n+1}}^1[n] + f_{T_{n+1}}^2[n+1] + \dots + f_{T_{n+1}}^z[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} = 0^* \\
&\Rightarrow \lim_{n \rightarrow \infty} \frac{f_{T_{n+1}}^i[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} = 0 \quad \forall i \\
&\quad \lim_{n \rightarrow \infty} \frac{f_{n+1}^i[n]}{O[n]} = 0 \quad \forall i
\end{aligned}$$

* $O[n]$ is a positive, non-decreasing function

25.3 Theorem of Polynomial Subfunctions

The Theorem of Polynomial Subfunctions states a solution has polynomial complexity if and only if all of its subfunctions have polynomial complexity

$$|s^+[X_n]| = O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n$$

$$s^+ = Sub_1[X_n] \cup Sub_2[X_n] \cup \dots \cup Sub_z[X_n]$$

$$O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n$$

$$\Longleftrightarrow$$

$$|Sub_i[X_n]| = O_i[n] = (\hat{\lambda}_M n)^M + (\hat{\lambda}_{M-1} n)^{M-1} + \dots + \hat{\lambda}_1 n + C \quad \forall i, n$$

25.3.1 Sufficient Proof

Solution s^+ having polynomial complexity implies all of its subfunctions Sub_i have polynomial complexity

Let

$$|s^+[X_n]| = O[n] = (\lambda_K n)^K + (\lambda_{K-1} n)^{K-1} \dots + \lambda_1 n + C \quad \forall n$$

$$O[n] = \sum_{i=1}^z O_i[n] = O_1[n] + O_2[n] + \dots + O_z[n]$$

Since $O[n], O_i[n]$ is positive, non-decreasing

$$O_i[n] = (\hat{\lambda}_{M_i} n)^{M_i} + (\hat{\lambda}_{M_i-1} n)^{M_i-1} \dots + \hat{\lambda}_1 n + C \quad M_i \leq K \quad \forall i, n$$

$\Rightarrow Sub_i$ has polynomial complexity by definition of polynomial complexity

25.3.2 Necessary Proof

Every subfunction Sub_i having polynomial complexity implies solution s^+ has polynomial complexity

Let

$$O_i[n] = (\hat{\lambda}_{M_i} n)^{M_i} + (\hat{\lambda}_{M_i-1} n)^{M_i-1} + \dots + \hat{\lambda}_{1_i} n + \hat{\lambda}_{0_i} \quad \forall i, n$$

$$O_{max}[n]^* := \tilde{O}[n] \in \{O_1[n], O_2[n], \dots, O_z[n]\} :$$

$$\lim_{n \rightarrow \infty} \frac{\tilde{O}[n]}{\sum_{i=1}^z O_i[n]} = c \neq 0$$

$$O_{max}[n] = (\hat{\lambda}_{M_{max}} n)^{M_{max}} + (\hat{\lambda}_{M_{max}-1} n)^{M_{max}-1} + \dots + \hat{\lambda}_{1_{max}} n + \hat{\lambda}_{0_{max}} \quad \forall i, n$$

$$O[n] = \sum_{i=1}^z O_i[n] = O_1[n] + O_2[n] + \dots + O_z[n]$$

$$= (\tilde{\lambda}_L n)^L + (\tilde{\lambda}_{L-1} n)^{L-1} + \dots + \tilde{\lambda}_1 n + C \quad L = M_{max} \quad \forall n$$

$\Rightarrow s^+$ has polynomial complexity by definition of polynomial complexity

* O_{max} is not necessarily unique, but necessarily exists. See appendix for proof

26 Divergent Solution Subfunction Properties

26.1 Restate Definition of Subfunction

$$\begin{aligned}
X_i &= \{x_1, \dots, x_n, C\}; \quad \hat{X}_i = \{x_1, \dots, x_{n+1}, C\} \\
s^+ &= s^+[X_i] := P : \\
(P[X_i] \rightarrow y_o == a_o \quad \forall X_i) &\cap (P[\hat{X}_i] \supseteq P[X_i] \quad \forall X_i, \hat{X}_i) \\
s^+ &= \{s_1, s_2, \dots, s_N | b_1, b_2, \dots, b_M, y_o\} = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} \\
&= \{\mathcal{L}, \mathcal{M}, y_o\} \\
Sub[X_i] &:= S = \{s_j, \dots | b_k, \dots, y_o\} : \\
s_j, b_k &\in s^+ \quad \forall s_j, b_k \in S
\end{aligned}$$

26.2 Property of Divergent Subfunctions

Let

$$D \in \hat{\mathcal{D}}$$

$$\begin{aligned}
X_n &= \{x_1, \dots, x_n\}; \quad \hat{X}_n = \{x_1, \dots, x_{n+1}\} \\
s^+ &= s^+[X_n] := P : \\
(P[X_i] \rightarrow y_o == a_o \quad \forall X_n) &\cap (P[\hat{X}_n] \supseteq P[X_n] \quad \forall \hat{X}_n : \hat{X}_n \supset X_n)
\end{aligned}$$

By Definition of Divergent Problem

$$\begin{aligned}
&\nexists c : \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} = c \\
&= \lim_{n \rightarrow \infty} \frac{O_T^1[n+1] + O_T^2[n+1] + \dots + O_T^z[n+1] + O_S[n+1]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} \\
&= \lim_{n \rightarrow \infty} \frac{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n] + f_{T_{n+1}}^1[n] + f_{T_{n+1}}^2[n+1] + \dots + f_{T_{n+1}}^z[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} \\
&= \lim_{n \rightarrow \infty} 1 + \frac{f_{T_{n+1}}^1[n] + f_{T_{n+1}}^2[n+1] + \dots + f_{T_{n+1}}^z[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} \neq c \\
&\Rightarrow \lim_{n \rightarrow \infty} \frac{f_{T_{n+1}}^1[n] + f_{T_{n+1}}^2[n+1] + \dots + f_{T_{n+1}}^z[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} \neq c^*
\end{aligned}$$

Prove

$$\exists i : \lim_{n \rightarrow \infty} \frac{f_{T_{n+1}}^i[n] + f_{S_{n+1}}[n]}{O_T^1[n] + O_T^2[n] + \dots + O_T^z[n] + O_S[n]} \text{ diverges}$$

* $O[n]$ is a positive, non-decreasing function

26.3 Theorem of Divergent Subfunctions

The Theorem of Divergent Subfunctions states a divergent subfunction implies divergent total complexity

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} \text{ diverges} \\ \iff \\ \exists i : \lim_{n \rightarrow \infty} \frac{O_i[n+1]}{O[n]} \text{ diverges} \end{aligned}$$

26.3.1 Sufficient Direction

See 18.2

26.3.2 Necessary Direction

27 Computational Basis

27.1 Definition of a Computational Basis of Program P

Define a Computational Basis B of Program P

$$X_n = \{x_1, x_2, \dots, x_n\}$$
$$P[X_n] \rightarrow Y_o := \{s_1, s_2, \dots, s_N, b_1, b_2, \dots, Y_o\} \rightarrow Y_o$$

$$B :=$$

For the remainder of this document, "computational basis" is denoted as "basis"

27.2 Definition of the Identity Basis of Program P

27.3 Prove the Identity Basis of Program P is a basis of Program P

27.4 Definition of Canonical Program

27.5 Definition of a Canonical Basis of Program P

27.6 Prove Canonical Basis \mathbb{B} of Program P is a basis of Program P

27.7 Subprogram and Canonical Basis

Prove a subprogram is a canonical basis if and only if it's basis decomposition is the identity subprogram

27.8 Basis of Boolean Program P

28 Fundamental Theorem of Computation

The Fundamental Theorem of Computation states every program P has a canonical basis \mathbb{B}

28.1 Proof

29 Input Spaces

29.1 Definition of Input Space

Define the Input Space \mathbb{I} of Program P

29.2 Define the Cardinality Function $C[n]$ of Input Space \mathbb{I}

29.3 Existence, Uniqueness, etc.

29.4 Worst Case

29.5 Prove $|\mathbb{B}| = C[n]$

30 Theorem of Solution Complexity

The Theorem of Solution Complexity relates the complexity of solution s^+ to a basis B of solution s^+

$$X_n = \{x_1, x_2, \dots, x_n\}$$

$$\mathcal{Q} := f[X_n] \rightarrow A_o \subseteq \Omega \quad \forall X_n \in D_{\mathcal{Q}}$$

$$s^+ = s^+[X_n] := P[X_n] \rightarrow Y_o :$$

$$(Y_o = A_o \quad \forall X_n \in D_{\mathcal{Q}}) \quad \cap \quad (P[X_{n+1}] \supseteq P[X_n] \quad \forall X_n \in D_{\mathcal{Q}} \quad \forall X_{n+1} \in D_{\mathcal{Q}})$$

31 Theorem of Optimal Complexity

32 Theorem of Divergent Complexity

33 Sum to N Problem with 2 integers

33.1 State formal definition of Sum to N : $x_i + x_j == N$

$$X_n = \{x_1, \dots, x_n\}$$

$$D := f[X_i, N] = a_o \in \{\mathbb{T}, \mathbb{F}\} \quad \forall X_i$$

$$s^+[X_n] = P[X_n] :$$

$$(P[X_i] = y_o == a_o \quad \forall X_i) \cap (P[X_{n+1}] \supseteq P[X_n] \quad \forall X_{n+1})$$

$$s^+ = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$D = f[X_i] = \exists x_j, x_k \in X_n \quad j \neq k :$$

$$x_j + x_k == N$$

33.2 Express a formal solution : $O_S[n] \sim n^0$

$$s^+ = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$s_1 = y_o \leftarrow \mathbb{F};$$

$$\forall i < n, \quad n \geq j > i$$

$$s_2, s_3, s_8, s_9, \dots, s_{3ij-4}, s_{3ij-3}, \dots, s_{3n(n-1)-4}, s_{3n(n-1)-3} = b_1 \leftarrow x_i + x_j$$

$$s_4, s_5, s_{10}, s_{11}, \dots, s_{3ij-2}, s_{3ij-1}, \dots, s_{3n(n-1)-2}, s_{3n(n-1)-1} = b_1 \leftarrow b_1 == N$$

$$s_6, s_7, s_{12}, s_{13}, \dots, s_{3ij}, s_{3ij+1}, \dots, s_{3n(n-1)}, s_{3n(n-1)+1} = y_o \leftarrow y_o \vee b_1$$

$$s^+ = \{y_o \leftarrow \mathbb{F}, y_o \leftarrow y_o \vee (x_i + x_j == N) \quad \forall i, j > i \mid b_1, y_o\}$$

33.3 Prove s^+ satisfies the subfunction condition of solutions:

$$P[X_{n+1}] \supseteq P[X_n] \quad \forall X_{n+1}$$

$$X_n = \{x_1, x_2, \dots, x_n\}; \quad X_{n+1} = \{x_1, x_2, \dots, x_n, x_{n+1}\}$$

$$s^+ = \{s_1, s_2, \dots, s_{O_T[n]}, b_1, b_2, \dots, b_{O_S[n]}, y_o\} = \{\mathcal{L}, \mathcal{M}, y_o\}$$

$$s_{n+1}^+ = s^+ \cup \hat{s}^+$$

$$s_1 = y_o \leftarrow \mathbb{F};$$

$$\forall i < n, \quad n \geq j > i$$

$$\begin{aligned}
s_2, s_3, s_8, s_9, \dots, s_{3ij-4}, s_{3ij-3}, \dots, s_{3n(n-1)-4}, s_{3n(n-1)-3} &= b_1 \leftarrow x_i + x_j \\
s_4, s_5, s_{10}, s_{11}, \dots, s_{3ij-2}, s_{3ij-1}, \dots, s_{3n(n-1)-2}, s_{3n(n-1)-1} &= b_1 \leftarrow b_1 == N \\
s_6, s_7, s_{12}, s_{13}, \dots, s_{3ij}, s_{3ij+1}, \dots, s_{3n(n-1)}, s_{3n(n-1)+1} &= y_o \leftarrow y_o \vee b_1
\end{aligned}$$

$$\begin{aligned}
\forall k &< n + 1 \\
s_{\dots} &= b_1 \leftarrow x_k + x_{n+1} \\
s_{\dots} &= b_1 \leftarrow b_1 == N \\
s_{\dots} &= y_o \leftarrow y_o \vee b_1
\end{aligned}$$

$$\begin{aligned}
s^+ &= \{y_o \leftarrow \mathbb{F}, y_o \leftarrow y_o \vee (x_i + x_j == N) \mid \forall i, j > i \mid b_1, y_o\} \\
\hat{s}^+ &= \{y_o \leftarrow y_o \vee (x_k + x_{n+1} == N) \mid \forall k < n + 1 \mid b_1, y_o\} \\
s_{n+1}^+ &= \{y_o \leftarrow \mathbb{F}, y_o \leftarrow y_o \vee (x_i + x_j == N) \mid \forall i, j > i \mid b_1, y_o\} \cup \\
&\quad \{y_o \leftarrow y_o \vee (x_k + x_{n+1} == N) \mid \forall k < n + 1 \mid b_1, y_o\} \\
s_{n+1}^+ &= s^+ \cup \hat{s}^+ = P[X_{n+1}] \supseteq P[X_n] = s^+
\end{aligned}$$

33.4 Determine $O[n]$, $O_S[n]$, $O_T[n]$, $f_{n+1}[n]$, $f_{n+1}^T[n]$, $f_{n+1}^S[n]$ for the above solution

$$\begin{aligned}
O_S[n] &= |y_o| + |b_1| = 2 \\
O_T[n] &= 3n(n-1) + 1 = 3n(n-1) - 1 + O_S[n] \\
O[n] &= 3n(n-1) + 3 = 3n^2 - 3n + 3 \\
f_{n+1}^S[n] &= 0 \\
f_{n+1}^T[n] &= 6n \\
f_{n+1}[n] &= f_{n+1}^S[n] + f_{n+1}^T[n]
\end{aligned}$$

33.5 Verify $O[n+1] = O[n] + f_{n+1}[n]$

$$\begin{aligned}
O[n+1] &= O[n] + \hat{O}[n] \\
3(n+1)^2 - 3(n+1) + 3 &= 3n^2 - 3n + 3 + 6n \\
3n^2 + 6n + 3 - 3n - 3 + 3 &= 3n^2 + 3n + 3
\end{aligned}$$

$$3n^2 + 3n + 3 = 3n^2 + 3n + 3$$

33.6 Show s^+ has Polynomial Complexity by the definition of Total Polynomial Complexity

$$O[n] = 3n^2 - 3n + 3$$

33.7 Show the limit $\lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]}$ does not Diverge

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{O[n+1]}{O[n]} &= \\ \lim_{n \rightarrow \infty} \frac{3n^2 + 3n + 3}{3n^2 - 3n + 3} &= \\ \lim_{n \rightarrow \infty} \left(\frac{3n^2 - 3n + 3}{3n^2 - 3n + 3} + \frac{6n}{3n^2 - 3n + 3} \right) &= \\ \lim_{n \rightarrow \infty} \left(1 + \frac{6n}{3n^2 - 3n + 3} \right) &= 1 \end{aligned}$$

34 The Knapsack Problem

34.1 The Knapsack Problem

The Knapsack Problem is a famous problem in computer science which asks if objects can be stored in a knapsack. Typically the problem is designed with two constraints, weight and value. Given objects x_i , each with a respective weight w_i and value v_i , does there exist a combination of objects lighter than input weight W and more valuable than input value V ?

34.2 Formal Definition

$$X_n = \{x_1, x_2, \dots, x_n\} = \{\{w_1, v_1\}, \{w_2, v_2\}, \dots, \{w_n, v_n\}\}$$

$$I = \{i_1, i_2, \dots, i_n\} : i_l \in \{0, 1\} \quad \forall i_l \in I$$

$$D := f[X_n, W, V] = a_o \in \{\mathbb{T}, \mathbb{F}\} = \exists I : \\ (\sum_{j=1}^n i_j w_j < W) \wedge (\sum_{j=1}^n i_j v_j \geq V)$$

34.3 Solve for $C[n]$

34.3.1 Expressing I as a binary number

$$I = \{i_1, i_2, \dots, i_n\} : i_l \in \{0, 1\} \quad \forall i_l \in I$$

Valid combinations of I

$$I_{valid} = \\ \{\{0, 0, 0, \dots, 0, 0, 1\}, \{0, 0, 0, \dots, 0, 1, 0\}, \{0, 0, 0, \dots, 0, 1, 1\}, \dots, \{1, 1, 1, \dots, 1, 1, 1\}\}$$

$$C[n] = |I_{valid}[n]| = 2^n - 1$$

34.3.2 Using a sum of combinations of inputs x_i

$$X_n = \{x_1, x_2, \dots, x_n\} = \{\{w_1, v_1\}, \{w_2, v_2\}, \dots, \{w_n, v_n\}\}$$

Valid combinations of x_i

$$\begin{aligned}
X_{valid}[n] &= \\
&\{x_1\} \cup \{x_2\} \cup \dots \cup \{x_n\} \cup \{x_1, x_2\} \cup \{x_1, x_3\} \cup \dots \cup \{x_{n-1}, x_n\} \cup \dots \cup \{x_1, x_2, \dots, x_n\} \\
&= {}_n C_1 \cup {}_n C_2 \cup \dots \cup {}_n C_n \\
C[n] &= |X_{valid}[n]| = \sum_{j=1}^n {}_n C_j
\end{aligned}$$

34.3.3 Verify consistency

$$\begin{aligned}
C[n] &= |X_{valid}[n]| = |I_{valid}[n]| \\
&= 2^n - 1 = \sum_{j=1}^n {}_n C_j = {}_n C_1 + {}_n C_2 + \dots + {}_n C_n \\
&= 2^n - 1 = 2^n - 1
\end{aligned}$$

34.4 Express a solution s^+ of the Knapsack Problem

34.5 Prove s^+ satisfies the subfunction condition of solutions

34.6 Determine $O[n], O_T[n], O_S[n], f_{n+1}[n]$

34.7 Show $s^+ \notin \mathbb{P}$

34.8 Express the Solution Space \mathbb{S} for The Knapsack Problem

34.9 Prove a lower bound for all solutions $s^+ \in S^+ := O_{lower}[n]$

34.10 Prove D has Divergent Complexity

Appendix

Types of Methods

34.11 Void Methods

Define a void program; a program with inputs x_i and no output

$$X_n = \{x_1, \dots, x_n\}$$
$$P = P[X] := \{s_1, s_2, \dots, s_N \mid b_1, b_2, \dots, b_M\}$$

34.12 Numerical Methods

Define a numerical program; a program with inputs x_i , input set C, and real, rational output y_o

$$X = \{x_1, \dots, x_n, C\}$$
$$P = P[X] := \{s_1, s_2, \dots, s_N \mid b_1, b_2, \dots, b_M, y_o\} =$$
$$P[X] \rightarrow y_o \in \mathbb{Q} \quad y_o \geq 0$$

34.13 System Methods

Define a system program; a program with inputs x_i , input set C, and real, output set Y_o

$$X = \{x_1, \dots, x_n, C\}$$
$$P = P[X] := \{s_1, s_2, \dots, s_N \mid b_1, b_2, \dots, b_M, Y_o\} =$$
$$P[X] \rightarrow Y_o = \{y_1, y_2, \dots, y_K\}$$

34.14 Mathematical Methods

Define a mathematical program; a program with inputs x_i , input set C and numerical output y_o

$$X = \{x_1, \dots, x_n, C\}$$
$$P = P[X] := \{s_1, s_2, \dots, s_N \mid b_1, b_2, \dots, b_M, y_o\} =$$
$$P[X] \rightarrow y_o \in \mathbb{Q}$$

34.15 While Loop

35 Criticism of Overloaded Equivalence

In computer science, it is convention to overload equivalence =

$$a_i = a_i \quad \forall a_i \in \Omega$$

Consider standard C++ syntax

```
int x = 3;  
int y = 4;  
int z = x + y;
```

Int x is not inherently equal to 3. Rather, we are creating an open space "x" for a value and setting the value to 3. Similarly, z is not inherently equal to the value of x + y. Rather, we are creating an open space "z" for a value and setting the value to the sum of x and y which have already been set.

$$\begin{aligned}x &\leftarrow 3 \\y &\leftarrow 4 \\z &\leftarrow x + y\end{aligned}$$

36 Existence of $O_{max}[n]$

36.1 Proof

36.1.1 Alternate Definition; Left Hand Derivative

Some sources define

$$\Delta_n^1 f[n] = f[n] - f[n-1]$$

Citations

- [1] https://kapeli.com/cheat_sheets/LaTeX_Math_Symbols.docset/Contents/Resources/Documents/
- [2] chat.openai.com
- [3] [google.com](https://www.google.com/)
- [4] [wikipedia.org](https://en.wikipedia.org/)
- [3] [wolframalpha.com](https://www.wolframalpha.com/)
- [4] [youtube.com](https://www.youtube.com/)
- [5] <https://stackoverflow.com/questions/3518973/floating-point-exponentiation-without-power-function>
- [6] <https://stackoverflow.com/questions/27086195/linear-index-upper-triangular-matrix>