KIT: Graph Theory

Ex. 5 Matroids

pz von p Z.

Beitreten

Vorschau

Let $E := \{1,3,5,9,11\}$ and $I := \{A \subseteq E \mid \sum_{e \in A} e \leq 20\}$. Prove that (E, I) is an independence set but not a matroid.

Author

p Z.

Informationen

Zuletzt geändert vor 2 Jahren

Kurs melden

1. Die leere Menge ist in der Menge der unabhängigen Mengen enthalten (diese Bedingung ist offensichtlich erfüllt, da die Summe der Elemente der leeren Menge 0 ist, was kleiner als 20 ist).
2. Wenn eine Menge B in I ist und A eine Teilmenge von B ist, dann ist auch A in I (diese Eigenschaft nennt man die Heritabilität oder "Vererbungseigenschaft").
3. Wenn A und B unabhängige Mengen sind und A mehr Elemente als B enthält, dann gibt es ein Element in A, das zu B hinzugefügt werden kann, so dass die resultierende Menge auch unabhängig ist (diese Eigenschaft nennt man die Austauscheigenschaft oder "Exchange-Eigenschaft").

1. $\emptyset \in I$
2. $A \in I$ and $B \subseteq A \Rightarrow B \in I$

$\boxed{A \in I}$, $\underbrace{B \subseteq A}$

$$\underbrace{\sum_{e \in B} e \le \sum_{e \in A} e}_{e \ge 0} \le 20$$

3. Können nichts aus B hinzufügen ohne, dass die 20 überschritten wird und wir dadurch gegen die Definition von der Independence verstoßen

$A, B \in I$ and $|A| > |B|$
$\Rightarrow \exists a \in A \setminus B$ s.t. $B \cup \{a\} \in I$

$B := \{\underbrace{9, 11}_{20}\}$ $\quad A := \{1, 2, 5, 9\}$

$|B| = 2$ $\quad |A| = 4$

$A \setminus B = \{1, 2, 5\}$

Exercise 2 (Matroid Properties)

Let (E, I) be an independence system. Prove that the following statements are equivalent:

- (E, I) is a matroid.
- For each subset $A \subseteq E$ it holds that if $M, M' \in I$ are maximal with respect to A, then $|M| = |M'|$.

Remark: A set M is called maximal with respect to A if the following holds:

- $M \subseteq A$
- $M \subseteq M' \subseteq A$ implies $M = M'$

Die neue Aufgabe, die Sie hochgeladen haben

Die neue Aufgabe, die Sie hochgeladen haben, behandelt eine andere Eigenschaft von Matroiden. Es wird ein Unabhängigkeitssystem (E, I) betrachtet und man soll beweisen, dass die beiden folgenden Aussagen äquivalent sind:

1. (E, I) ist ein Matroid.
2. Für jede Teilmenge A ⊆ E gilt, dass wenn M und M' in I sind und maximal bezüglich A, dann haben M und M' die gleiche Mächtigkeit (Anzahl der Elemente), d.h. $|M| = |M'|$.

Eine Menge M wird als maximal bezüglich einer Teilmenge A bezeichnet, wenn M eine unabhängige Menge in I ist, die vollständig in A enthalten ist, und es keine größere unabhängige Menge gibt, die auch vollständig in A enthalten ist.

Um die Äquivalenz dieser Aussagen zu veranschaulichen, kann man das folgende einfache Beispiel verwenden:

Betrachten wir ein Unabhängigkeitssystem (E, I), wobei E = {1, 2, 3, 4} und die unabhängigen Mengen I alle Teilmengen von E sind, die nicht mehr als zwei Elemente enthalten.

Beispiel:

- Für die Teilmenge A = {1, 2, 3} von E könnte eine maximale unabhängige Menge M = {1, 2} sein. Eine andere maximale unabhängige Menge M' = {2, 3} ist auch möglich.
- Beide Mengen M und M' sind maximal bezüglich A, weil das Hinzufügen eines weiteren Elements von A die Menge nicht mehr unabhängig machen würde (da die Mengen in I definiert sind, um nicht mehr als zwei Elemente zu haben).
- In diesem Beispiel haben M und M' die gleiche Mächtigkeit, nämlich |M| = |M'| = 2.

Für das gegebene System (E, I) können wir zeigen, dass die Eigenschaften eines Matroids erfüllt sind:

- Es gibt eine Austauscheigenschaft, die besagt, dass, wenn eine unabhängige Menge M mehr Elemente als eine andere unabhängige Menge M' hat, es möglich ist, ein Element von M zu M' hinzuzufügen und M' bleibt unabhängig. Da alle maximalen unabhängigen Mengen bezüglich einer Teilmenge A die gleiche Größe haben, wird diese Bedingung immer erfüllt sein.
- Daraus folgt, dass die Eigenschaft, dass alle maximalen unabhängigen Mengen bezüglich einer Teilmenge A die gleiche Mächtigkeit haben, äquivalent dazu ist, dass (E, I) ein Matroid ist.

Diese Übung demonstriert eine der grundlegenden Eigenschaften von Matroiden, nämlich dass in einem Matroid alle maximalen unabhängigen Mengen bezüglich einer beliebigen Teilmenge die gleiche Mächtigkeit haben müssen. Dies ist eine direkte Folge der Austauscheigenschaft von Matroiden.

- Let (E, I) be an independence system. Prove that the following statements are equivalent:

- (E, I) is a matroid. A
- For each subset $A \subseteq E$ it holds that if $M, M' \in I$ are maximal with respect to A, then $|M| = |M'|$. B

- **Remark**: A set M is called maximal with respect to A if the following holds:
  - $M \subseteq A$ and $M \subseteq M' \subseteq A$ implies $M = M'$

A $\Rightarrow$B | $\bar{B} \Rightarrow \bar{A}$

- $\Leftarrow$ Show that from B not being fulfilled it must be that A is not fulfilled either.
  - Suppose M, M' are maximal with respect to A and $|M| < |M'|$. } B
  - For such two sets M, M' , the matroid property
    - $X, Y \in I$ and $|X| < |Y| \Rightarrow \exists y \in Y \backslash X$ s.t. $X \cup \{y\} \in I$
  - requires that as $M, M' \in I$ and $|M| < |M'|$ it follows that $\exists e \in M' \backslash M$ s.t. $M \cup \{e\} \in I$.
  - As $M \subseteq A$ and $e \in A$, $M + e \in A$ which contradicts the maximality of M in A.

$M \cup \{e\}$

Graph Theory and Advanced Location Models   Institute of Operations Research

B $\Rightarrow$A

- $\Rightarrow$ Show that from B being fulfilled, A must also be fulfilled.
  - Let $A, B \in I$, with $|B| < |A|$.
  - Let $X := A \cup B$.
  - Since $A \in I$, there exists a maximal independent set M_A with respect to X with $A \subseteq M_A$.
  - Similarly, let M_B be a maximal independent set with respect to X with $B \subseteq M_B$.
  - It follows that $|B| < |A| \leq |M_A|$.
  - Since according to B all maximal sets w.r.t. X have the same cardinality, B cannot be maximal w.r.t. X. Thus, $\exists e \in M_B \backslash B \mid B + e \in I$
    - $B + e \subseteq M_B$ and $M_B \in I$, $B + e$ is independent.
  - Since M_B only contains elements from $A \cup B$ it follows that $\exists a \in A \backslash B \mid B + a \in I$.

Exercise 3: Matroids and the Greedy Algorithm

- Let (E, I) be a matroid with weights on E and assume that the elements of $E = \{e_1, e_2, \ldots, e_m\}$ are sorted by decreasing weight. Prove the greedy-choice property of matroids: If e_1 is such that e_1 is independent, then there exists an optimal subset A of E containing e_1.

Die Aufgabe bezieht sich auf die Anwendung des Greedy-Algorithmus (Gierigen Algorithmus) auf Matroide. Das Ziel ist es zu beweisen, dass wenn ein Element e1 in einem gewichteten Matroid (E, I) unabhängig ist, dann existiert eine optimale Teilmenge A von E, die e1 enthält.

Hier sind die Schritte, um dies zu beweisen, zusammen mit einem einfachen Beispiel:

1. **Annahme**: Die Elemente von E = {e1, e2, ..., em} sind nach abnehmendem Gewicht sortiert. Das bedeutet, dass e1 das Element mit dem höchsten Gewicht ist.

2. **Unabhängigkeit von e1**: Wir nehmen an, dass {e1} eine unabhängige Menge ist. Das bedeutet, dass es eine Teilmenge von E gibt, die nur aus e1 besteht und eine gültige unabhängige Menge in I ist.

3. **Existenz einer optimalen Teilmenge**: Wir wollen beweisen, dass es eine optimale Teilmenge A von E gibt, die e1 enthält. Eine optimale Teilmenge ist eine, die das größtmögliche Gesamtgewicht unter allen unabhängigen Mengen hat.

4. **Verwendung des Greedy-Algorithmus**: Der Greedy-Algorithmus würde damit beginnen, e1 zu wählen, weil es das Element mit dem höchsten Gewicht ist. Danach würde der Algorithmus weitere Elemente in abnehmender Gewichtsreihenfolge hinzufügen, solange die resultierende Menge unabhängig bleibt.

5. **Optimalität**: Angenommen, es gibt eine optimale Teilmenge A∗, die e1 nicht enthält. Da e1 das Element mit dem höchsten Gewicht ist, könnte man e1 zu A∗ hinzufügen, wenn A* nicht bereits maximal ist. Wenn A* maximal ist, könnte man ein Element aus A* entfernen und e1 hinzufügen, um eine neue unabhängige Menge zu bilden, die mindestens so schwer wie A* ist. Dies würde bedeuten, dass A∗ nicht optimal sein kann, wenn sie nicht e1 enthält.

Einfaches Beispiel:

Angenommen, wir haben ein Matroid (E, I), wobei E = {a, b, c, d} und die Gewichte sind wie folgt: Gewicht(a) > Gewicht(b) > Gewicht(c) >

Gewicht(d). Die Menge I besteht aus allen Teilmengen von E, die zwei oder weniger Elemente enthalten.

Nun nehmen wir an, dass {a} eine unabhängige Menge ist, da sie nur ein Element enthält und die Bedingung von I erfüllt.

Der Greedy-Algorithmus würde beginnen, indem er a wählt, weil es das Element mit dem höchsten Gewicht ist. Danach könnte er b hinzufügen, um die Menge {a, b} zu bilden, die immer noch unabhängig ist.

Wenn es eine andere optimale Teilmenge A∗ gäbe, die a nicht enthält, könnte sie beispielsweise {b, c} sein. Wir könnten dann aber a zu dieser Menge hinzufügen, um {a, b, c} zu bilden, was gegen die Regeln von I verstößt, da I nur Mengen mit bis zu zwei Elementen erlaubt. Stattdessen könnten wir c durch a ersetzen, um {a, b} zu bilden, was eine unabhängige Menge ist und ein höheres Gesamtgewicht als A* hat, wenn A* a nicht enthält. Daher muss jede optimale Teilmenge a enthalten.

Exercise 3: Matroids and the Greedy Algorithm

- If no such e_1 exists, then the only independent subset is the empty set and the theorem is trivially true
- Let B be any nonempty optimal subset and assume $\boxed{e_1} \notin B$, otherwise the theorem is already fulfilled for $A = B$
- We have that no element of B has weight greater than $w(e_1)$ because $w(e_1) \geq w(e)$ for any $e \in B \subset E$
- Construct A as follows:
  - Begin with $\boxed{A = \{e_1\}}$
  - Using the exchange property repeatedly find a new element of B that we can add to A until $|A| = |B|$, while preserving independence of A
  - At that point, A and B are the same except A has e_1 and B has some other element e: $A = (B - \{e\}) \cup \{e_1\}$ for some $e \in B$ and $w(A) = w(B) - w(e) + w(e_1) \geq w(B)$
  - B is optimal
    - Case 1: $-w(e) + w(e_1) = 0$: A containing e_1 must also be optimal
    - Case 2: $-w(e) + w(e_1) > 0$: B with $e_1 \notin B$ cannot be optimal → Contradiction → any optimal subset (in this case) must contain e_1

- If no such e_1 exists, then the only independent subset is the empty set and the theorem is trivially true

- Let B be any nonempty optimal subset and assume $e_1 \notin B$, otherwise the theorem is already fulfilled for $A = B$
- We have that no element of B has weight greater than $w(e_1)$ because $w(e_1) \geq w(e)$ for any $e \in B \subset E$
- Construct A as follows:
  - Begin with $A = \{e_1\}$
  - Using the exchange property repeatedly find a new element of B that we can add to A until $|A| = |B|$, while preserving independence of A
  - At that point, A and B are the same except A has e_1 and B has some other element e: $A = (B - \{e\}) \cup \{e_1\}$ for some $e \in B$ and $w(A) = w(B) - w(e) + w(e_1) \geq w(B)$
  - B is optimal
    - Case 1: $-w(e) + w(e_1) = 0$: A containing e_1 must also be optimal
    - Case 2: $-w(e) + w(e_1) > 0$: B with $e_1 \notin B$ cannot be optimal → Contradiction → any optimal subset (in this case) must contain e_1

Die Lösung, die Sie hochgeladen haben, ist ein formaler Beweis der Aussage, dass in einem gewichteten Matroid, wenn ein Element e_1 unabhängig ist, dann gibt es eine optimale Teilmenge, die e_1 enthält. Hier ist eine detaillierte Erklärung der Schritte und ein einfaches Beispiel:

Schritt 1: Trivialer Fall

- Wenn kein solches unabhängiges Element e_1 existiert, dann ist die einzige unabhängige Teilmenge die leere Menge, und die Behauptung ist trivialerweise wahr.

Schritt 2: Annahme einer optimalen Teilmenge ohne e_1

- Angenommen, es gibt eine optimale Teilmenge B, die e_1 nicht enthält. Da e_1 das Element mit dem größten Gewicht ist und unabhängig ist, muss jedes Element in B ein geringeres Gewicht als e_1 haben.

Schritt 3: Konstruktion von A

- Beginne mit $A=\{e_1\}$.
- Nutze die Austauscheigenschaft des Matroids, um Elemente aus B zu A hinzuzufügen, ohne die Unabhängigkeit von A zu verlieren, bis $|A|=|B|$.
- An diesem Punkt unterscheiden sich A

und B nur durch ein Element, d.h., A hat $e1$ und B hat ein anderes Element e, und es gilt $w(A) \geq w(B)$.

Schritt 4: Widerspruch und Schlussfolgerung

- Da B optimal ist, gibt es zwei Fälle zu betrachten:
    - **Fall 1**: Wenn das Entfernen von e aus B und das Hinzufügen von $e1$ zu B das gleiche Gewicht ergibt, dann ist auch A optimal.
    - **Fall 2**: Wenn A ein größeres Gewicht als B hat, dann kann B nicht optimal sein, was ein Widerspruch ist. Also muss jede optimale Menge $e1$ enthalten.

Einfaches Beispiel:

Nehmen wir an, wir haben ein Matroid mit der Elementmenge $E=\{x,y,z\}$ und Gewichten $w(x)=3, w(y)=2, w(z)=1$. Die unabhängigen Mengen sind alle Teilmengen von E mit höchstens zwei Elementen.

- $e1$ ist hier x, da es das höchste Gewicht hat und unabhängig ist (da jede einzelne Menge in diesem Matroid unabhängig ist).
- Angenommen, B ist die optimale Teilmenge $\{y,z\}$ mit einem Gesamtgewicht von 3.
- Wir beginnen mit $A=\{x\}$ mit einem Gewicht von 3.
- Wir können y nicht zu A hinzufügen, ohne die Unabhängigkeitsregel zu verletzen, da wir bereits zwei Elemente haben.
- Daher ist A auch eine optimale Teilmenge, die $e1$ enthält, da sie das gleiche Gewicht

wie B*B* hat, aber mit weniger Elementen.

Der Beweis zeigt, dass, wenn man mit dem Element mit dem höchsten Gewicht beginnt und die Austauscheigenschaft nutzt, man am Ende eine optimale Teilmenge konstruieren kann, die dieses Element enthält. Dies ist der Kern der Greedy-Choice-Eigenschaft bei Matroiden.

Exercise 4: Matroid Example and Application

- Consider n jobs which can be processed on a machine. Each job has processing time $p_j = 1$, but job-specific deadlines $d_j \in \{1,2,\dots\}$ and rewards r_j. The reward r_j for job j is only given when it completes processing on time (i.e., not after d_j). The overall problem consists of ordering the jobs such that the sum of given rewards is maximized.

a) Prove that (E, I) with E being the set of all jobs ($E = \{1,2,\dots,n\}$) and I being the set of all job sets which can be completed on time (i.e., each job of the job set can be completed on time) is a matroid.

- You can use the following lemmata and notation:
  - **Lemma 1**: If a job set can be completed on time, then this job set is also completed on time when its elements are ordered non-decreasingly by deadline.
  - For $t = 0, 1, 2, \dots, n$, let $N_t(A)$ denote the number of jobs in a given job set A whose deadline is t or earlier.
  - **Lemma 2**: For any job set A consisting of n jobs, the following statements are equivalent:
    - All jobs in A can be completed in time.
    - For $t = 0, 1, 2, \dots, n,$, we have $N_t(A) \leq t$.
    - If jobs in A are scheduled non-decreasingly by deadlines, then no job of A is late.

Um zu beweisen, dass die Struktur (E, I), die in der Aufgabe definiert ist, ein Matroid ist, müssen wir die drei grundlegenden Eigenschaften von Matroiden überprüfen:

1. Die leere Menge ist in I.
2. Jede Teilmenge einer Menge in I ist ebenfalls in I (Heritabilität oder Vererbung).

3. Wenn es zwei Mengen A und B in I gibt, wobei |A| < |B|, dann gibt es ein Element in B, das zu A hinzugefügt werden kann, so dass die resultierende Menge auch in I ist (Austauscheigenschaft).

Die gegebene Aufgabe liefert zwei Lemmata, die uns helfen, die dritte Eigenschaft zu beweisen.

Lemma 1: Ein Jobset kann rechtzeitig fertiggestellt werden, wenn seine Jobs in nicht abnehmender Reihenfolge nach Deadlines geordnet sind.

Lemma 2: Für jedes Jobset A, das aus n Jobs besteht, sind die folgenden Aussagen äquivalent:

- Alle Jobs in A können rechtzeitig abgeschlossen werden.
- Für alle t = 0, 1, 2, ..., n gilt, dass Nt(A)$Nt(A)$ ≤ t, wobei Nt(A)$Nt(A)$ die Anzahl der Jobs in A ist, deren Deadline t oder früher ist.
- Wenn Jobs in A nicht abnehmend nach Deadlines geplant sind, ist kein Job verspätet.

Matroid ist, weil:

1. **Die leere Menge ist offensichtlich in I**, da keine Jobs zu erledigen sind und somit keine Deadlines verpasst werden können.
2. **Heritabilität**: Jede Teilmenge eines Jobsets, das rechtzeitig fertiggestellt werden kann, kann auch rechtzeitig fertiggestellt werden, da das Entfernen von Jobs nicht dazu führt, dass verbleibende Jobs verspätet sind.
3. **Austauscheigenschaft**: Angenommen, wir

3. **Austauscheigenschaft**: Angenommen, wir haben zwei Jobsets A und B, die rechtzeitig fertiggestellt werden können, und |A| < |B|. Nach Lemma 2 haben wir Nt(A)$Nt(A)$ ≤ t und Nt(B)$Nt(B)$ ≤ t für alle t. Da |A| < |B|, gibt es einen Zeitpunkt t, bei dem Nt(B)>Nt(A)$Nt(B)>Nt(A)$, was bedeutet, dass es einen Job in B gibt, der eine Deadline von t oder früher hat, aber nicht in A ist. Dieser Job kann zu A hinzugefügt werden, ohne die Deadlines zu verletzen, was die Austauscheigenschaft bestätigt.

Diese Lemmata zeigen, dass die Struktur (E, I), wo E alle Jobs und I alle Jobsets sind, die rechtzeitig abgeschlossen werden können, ein Matroid ist, weil:

1. **Die leere Menge ist offensichtlich in I**, da keine Jobs zu erledigen sind und somit keine Deadlines verpasst werden können.

2. **Heritabilität**: Jede Teilmenge eines Jobsets, das rechtzeitig fertiggestellt werden kann, kann auch rechtzeitig fertiggestellt werden, da das Entfernen von Jobs nicht dazu führt, dass verbleibende Jobs verspätet sind.

3. **Austauscheigenschaft**: Angenommen, wir haben zwei Jobsets A und B, die rechtzeitig fertiggestellt werden können, und |A| < |B|. Nach Lemma 2 haben wir Nt(A)$Nt(A)$ ≤ t und Nt(B)$Nt(B)$ ≤ t für alle t. Da |A| < |B|, gibt es einen Zeitpunkt t, bei dem

gibt es einen Zeitpunkt t, bei dem Nt(B)>Nt(A)$Nt(B)>Nt(A)$, was bedeutet, dass es einen Job in B gibt, der eine Deadline von t oder früher hat, aber nicht in A ist. Dieser Job kann zu A hinzugefügt werden, ohne die Deadlines zu verletzen, was die Austauscheigenschaft bestätigt.

- Exchange / extension property: $\exists a_i$ 6
  - Suppose $A, B \in I$ with $|A| < |B|$
  - Let k be the largest t such that $N_t(B) \leq N_t(A)$
  - Such a value of t exists, since $N_0(A) = N_0(B) = 0$
  - Since $N_n(B) = |B|$ and $N_n(A) = |A|$, but $|B| > |A|$, we must have $k < n$ and $N_j(B) > N_j(A)$ for $j \in \{k+1, \dots, j, \dots, n\}$
  - Therefore, B contains more jobs with deadline $k+1$ than A does
  - Let a_i be a job in $B - A$ with deadline $k+1$ and let $A' = A \cup \{a_i\}$
  - We show that $A' \in I$ (i.e., A' can be completed on time) by the second property of Lemma 2
    - For $t \in \{0, 1, \dots, k\}$, we have $N_t(A') = N_t(A) \leq t$ since A is independent, i.e., $A \in I$
    - For $t \in \{k+1, \dots, n\}$, we have $N_t(A') \leq N_t(B) \leq t$ since B is independent, i.e., $B \in I$
    - Therefore, A' is independent , i.e., $A' \in I$

b) Solve the following problem instance: $n = 7$, $d = (d_1, \dots, d_7) = (1, 4, 6, 4, 2, 4, 3)$, $r = (r_1, \dots, r_7) = (30, 20, 10, 70, 60, 50, 40)$.

- b) Solve the following problem instance

Job	1	2	3	4	5	6	7
Deadline	1	4	6	4	2	4	3
Reward	30	20	10	70	60	50	40

  - Because we have a matroid, we can apply the greedy algorithm and we will obtain a optimal solution
  - Sorting by non-increasing rewards leads to the following table:

Job	4	5	6	7	1	2	3
Deadline	4	2	4	3	1	4	6
Reward	70	60	50	40	30	20	10

Exercise 4: Matroid Example and Application

- $X = \{4\}$ would lead to $N_0(X) = N_1(X) = 0$ → ok
- $X := \{4\}$, schedule [4]
- $X = \{4,5\}$ would lead to $N_0(X) = N_1(X) = 0,\ N_2(X) = 1$ → ok
- $X := \{4,5\}$, schedule [5, 4]
- $X = \{4,5,6\}$ would lead to $N_0(X) = N_1(X) = 0,\ N_2(X) = N_3(X) = 1$ → ok
- $X := \{4,5,6\}$, schedule [5, 4, 6]
- $X = \{4,5,6,7\}$ would lead to $N_0(X) = N_1(X) = 0,\ N_2(X) = 1, N_3(X) = \underline{2},\ N_4(X) = 4$ → ok
- $X := \{4,5,6,7\}$, schedule [5, 7, 4, 6]
- $X = \{4,5,6,7,1\}$ would lead to $N_0(X) = 0, N_1(X) = 1,\ N_2(X) = 2, N_3(X) = 3, \boxed{N_4(X) = 5}$ → not ok, X unchanged
- $X = X = \{4,5,6,7,2\}$ would lead to $N_0(X) = N_1(X) = 0,\ N_2(X) = 1, N_3(X) = 2, \boxed{N_4(X) = 5}$ → not ok, X unchanged
- $X = X = \{4,5,6,7,3\}$ would lead to $N_0(X) = N_1(X) = 0,\ N_2(X) = 1, N_3(X) = 2,\ N_4(X) = 4,$ $\boxed{N_5(X) = 4}$ → ok
- $X := \underline{\{4,5,6,7,3\}}$, **schedule [5, 7, 4, 6, 3] with weight 230 = 70+60+50+40+10**

Beispiel mit Erklärung:

Angenommen, wir haben folgende Jobs und Belohnungen (nach Sortierung):

1. Job 4: Deadline 4, Belohnung 70
2. Job 5: Deadline 2, Belohnung 60
3. Job 6: Deadline 4, Belohnung 50
4. Job 7: Deadline 3, Belohnung 40
5. Job 1: Deadline 1, Belohnung 30
6. Job 2: Deadline 4, Belohnung 20
7. Job 3: Deadline 6, Belohnung 10

Anwendung des Greedy-Algorithmus:

1. Wir fügen Job 4 hinzu (X = {4}). Keine Deadline wurde verpasst, N0(X)=N1(X)=0$N0(X)=N1(X)=0$, N4(X)=1$N4(X)=1$, alles ist in Ordnung.
2. Als nächstes fügen wir Job 5 hinzu (X = {4, 5}). N0(X)=N1(X)=0$N0(X)=N1(X)=0$, N2(X)=1$N2(X)=1$, was okay ist, weil die Anzahl der Jobs, die bis zur Deadline 2 abgeschlossen sein müssen, nicht größer

als 2 ist.

3. Fügen wir Job 6 hinzu (X = {4, 5, 6}). N3(X)=1$N3(X)=1$, N4(X)=2$N4(X)=2$, was immer noch in Ordnung ist.

4. Job 7 wird hinzugefügt (X = {4, 5, 6, 7}). N3(X)=2$N3(X)=2$, N4(X)=3$N4(X)=3$, was immer noch funktioniert, weil die Anzahl der Jobs, die bis zur Deadline 4 abgeschlossen sein müssen, nicht größer als 4 ist.

5. Fügen wir nun Job 1 hinzu (X = {4, 5, 6, 7, 1}), es zeigt sich, dass N1(X)$N1(X)$ größer als 1 wäre, was bedeutet, dass wir nicht beide Jobs (4 und 1) bis zur Deadline 1 abschließen können. Daher wird Job 1 nicht hinzugefügt.

6. Versuchen wir Job 2 hinzuzufügen (X = {4, 5, 6, 7, 2}), wir sehen wieder, dass N4(X)$N4(X)$ größer als 4 wäre, was nicht funktioniert. Daher wird Job 2 nicht hinzugefügt.

7. Schließlich fügen wir Job 3 hinzu (X = {4, 5, 6, 7, 3}). Es gibt keine Überschreitung der Deadlines, alle Jobs können rechtzeitig abgeschlossen werden, also ist X = {4, 5, 6, 7, 3} der finale Plan.