# Übung: Projekt

Hinweis: Beachten Sie zur Bearbeitung der nachfolgenden Aufgabenstellungen das Informationsmaterial

*L2\_3 Info Array.docx*

*L2\_3 Schuelerdaten Projekt.py*

**(I) Problemstellung**

Das Beratungsunternehmen Cheap\_IT berät Kunden bei der Umsetzung von smarten IT-Lösungen. Dabei ist das Unternehmen in Projekten organisiert, die eine fiktive Bezeichnung haben. Bei den Projekten werden die bisherigen Ausgaben gespeichert sowie das Gesamtbudget (vgl. (4) UML-Diagramm). Alle Projekte werden in dem Array projekte gespeichert (vgl. L2\_3 Schuelerdaten Projekt.py).

1. Öffnen Sie das Programm L2\_3 Schuelerdaten Projekt.py
2. Implementieren Sie im Hauptprogramm Algorithmen, die die folgenden Aufgabenstellungen erfüllen:
   1. Es sollen die niedrigsten Ausgaben ausgegeben werden.
   2. Es sollen die höchsten Ausgaben ausgegeben werden.
   3. Alle Budgetüberschreitungen sollen aufaddiert und ausgegeben werden.
3. Testen Sie Ihr Programm mit der entsprechenden Codierung im Hauptprogramm.

**(II) Problemanalyse**

1. Welche Ausgabedaten will man erhalten?

Höchste Ausgaben, niedrigste Ausgaben, gesamte Budgetüberschreitung

1. Welche Daten werden zur Bearbeitung benötigt?

Budgets der Projekte, Ausgaben der Projekte

1. Welche Eigenschaften haben die Eingabe-, Verarbeitungs- und Ausgabedaten? (**Variablenliste**)

|  |  |  |
| --- | --- | --- |
| **Bedeutung** | **Typ** | **Variable** |
| Höchste Ausgaben | Dezimalzahl | max\_a |
| Niedrigste Ausgaben | Dezimalzahl | min\_a |
| Gesamte Budgetüberschreitung | Dezimalzahl | ges\_budget\_ub |
| Budgets der Projekte | Dezimalzahl | budget |
| Ausgaben der Projekte | Dezimalzahl | ausgaben |

1. UML-Diagramm:



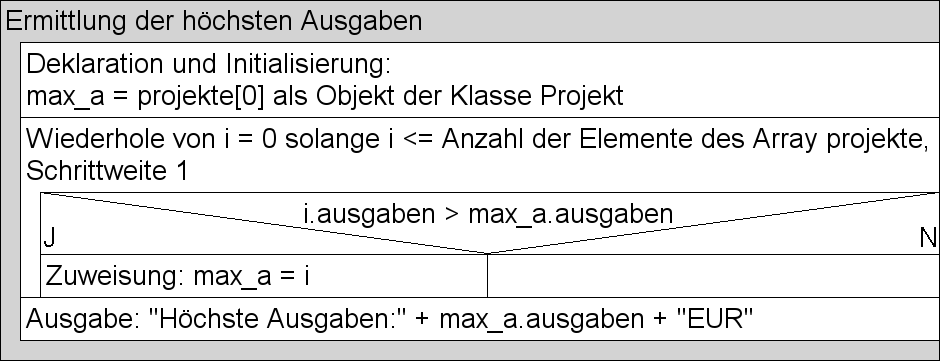
1. So soll die Bildschirmausgabe des Programms aussehen (Beispieldaten):

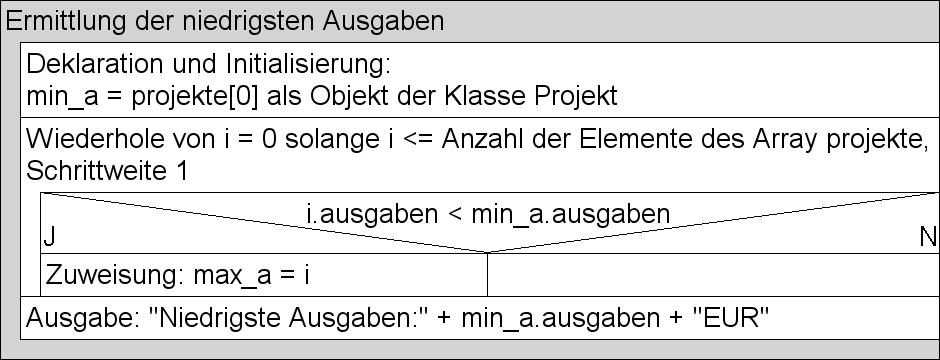
|  |
| --- |
| Höchste Ausgaben: 50000.0 EUR  Niedrigste Ausgaben: 5000.0 EUR  Gesamte Bugetüberschreitung: 3500.0 EUR |

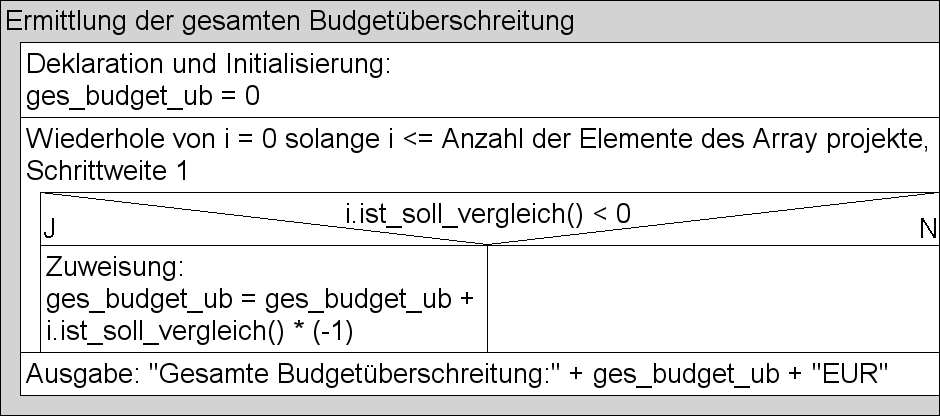
1. Verarbeitung

|  |
| --- |
| Höchste Ausgaben:  Für jedes Objekt  WENN i.ausgaben > max\_a.ausgaben  DANN max\_a = i  Niedrigste Ausgaben:  Für jedes Objekt  WENN i.ausgaben < min\_a.ausgaben  DANN min\_a = i  Gesamte Budgetüberschreitung:  Für jedes Objekt:  WENN i.ist\_soll\_vergleich() < 0  DANN ges\_budget\_ub = ges\_budget\_ub + i.ist\_soll\_vergleich() \* (-1) |

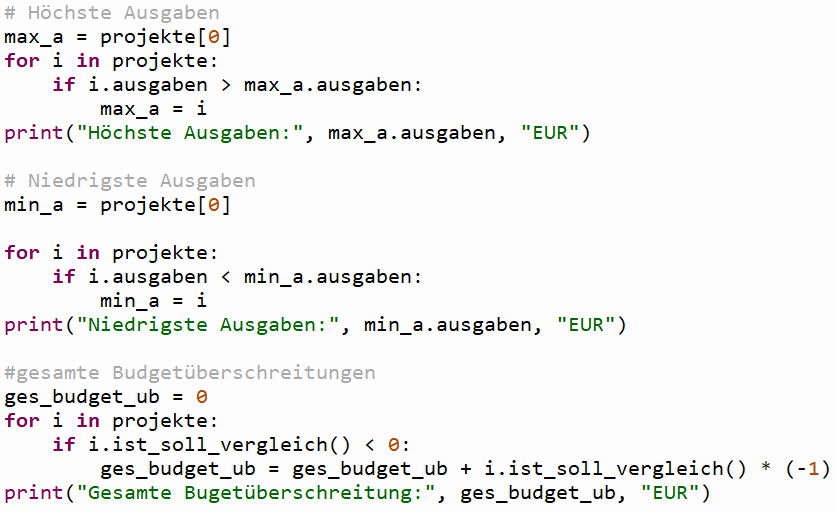
**(III) Struktogramm**

****

****

****

**(IV) Programmcode (Python-Code)**



**Datei:** L2\_3 Loesung Projekt.py