

BTS SN 2^{ème} ANNÉE – Option IR

IR2NUM1

LANGAGE C++ LES STRUCTURES

Durée : 6h

2020-2021

Introduction

La programmation orientée objet (la POO) et basée sur l'utilisation d'objets regroupant des données et des fonctions. Les structures maintenant remplacées par les classes sont un début de programmation orientée objet. Nous allons dans cette séance de travaux pratiques :

- **Apprendre à définir un modèle de structure et d'énuméré** : définir une structure type et un énuméré comportant des données diverses.
- **Manipuler des structures** : alimenter les données de plusieurs structures et les restituer.
- **Mettre en fonction** : Utiliser des fonctions pour automatiser la saisie et la restitution des données.

Dans chaque partie, il vous sera demandé de réaliser une certaine tâche professionnelle. A chaque fois, en conclusion de votre travail, vous devrez :



- Montrer au professeur que le **cahier des charges** a bien été rempli et répondre à ses questions ;
- Répondre aux questions du **document-réponse**.

Seront pris en compte dans l'évaluation de votre travail :

- La bonne réalisation des **installations** ;
- Le **soin** accordé au matériel ;
- Le bon **rangement** du matériel en fin de TP ;
- La **clarté** des explications données au professeur ;
- La **qualité rédactionnelle** du document-réponse.

En fin du TP, vous veillerez à :

- **Ranger** soigneusement le matériel ;
- Modifier le nom du document-réponse, en remplaçant les « **YYY** » par vos noms ;
- **L'envoyer** par mail ou le glisser dans la BAL de votre professeur.

Matériel disponible :



- Ordinateur équipé de *Windows 10*, et des logiciels *CodeBlock* ;
- Baies informatiques câblées et équipées de switches ;
- Câbles RJ45.

Ressources disponibles :



- Cours sur langage C ;
- « *Guide du Technicien* ».

Introduction

1. DEFINITION D'UNE STRUCTURE

2. INSTANTIATION DE STRUCTURE

3. UTILISATION DES STRUCTURES

4. FINALISATION DU PROGRAMME

5. Annexes

1. DEFINITION D'UNE STRUTURE

En vous servant de l'annexe créez un nouveau projet de type "Qt Console application" puis ajoutez un fichier d'entête nommé TP_structure.h au sein de votre projet. Dans ce fichier d'entête définissez deux types énumérés et deux types de structure qui soient conforme aux définitions ci-dessous :



Un type énuméré nommé **typeCouleur** avec comme valeur possible : BLEU, BLANC, VERT, ROUGE, VIOLET, NOIR, GRIS, JAUNE, MARRON.

Un type énuméré nommé **typeMotorisation** avec comme valeur possible : Diesel, Essence, GPL, Electrique.

Une structure nommée voiture comportant :

- Une chaine de caractère nommée **marque** de 50 caractères.
- Une chaine de caractère nommée **type** de 30 caractères.

- Une variable nommée **motorisation** de type typeMotorisation
- Un entier nommé **annee**
- Un entier long nommé **kilometrage**.
- Une variable nommée **couleur** de type typeCouleur

Une structure nommée garage comportant :

- Une chaine de 30 caractères nommée **nom**
- Une chaine de 100 caractères nommée **adresse**
- Un entier nommé **nombreMecanicien**
- Un tableau de 20 éléments d'objets de type voiture nommé **voitureDansGarage**.



- Complétez le compte-rendu de mesures.

2. INSTANTIATION DE STRUCTURE

Dans le fichier main.cpp de votre projet, instanciez deux variables de type garage que vous nommerez garage1 et garage2.



Garage1



+

Garage2



- Complétez le compte-rendu de mesures.

Introduction

1. DEFINITION D'UNE
STRUCTURE2. INSTANTIATION
DE STRUCTURE3. UTILISATION DES
STRUCTURES4. FINALISATION DU
PROGRAMME

5. Annexes

3. UTILISATION DES STRUCTURES

Suivez les principales étapes suivantes :



- Créez un programme qui interroge l'utilisateur et lui demande de saisir les informations sur un garage. Ces informations comportent aussi celles relatives à deux voitures nommées voiture1 et voiture2 et stockées dans le tableau voitureDansGarage.
- Transformez ce programme en une fonction nommée saisieGarage() qui sera appelée dans le programme principal.
- Créez une fonction nommée afficheGarage() qui affiche toutes les informations sur le garage et les voitures qui se trouvent dedans.



- Complétez le compte-rendu de mesures.

4. FINALISATION DU PROGRAMME

Suivez les principales étapes suivantes :



- Transformez les variables nommées garage1 et garage2 en un tableau nommé tousLesGarage de 20 objets de type garage.
- Transformez la fonction nommée saisieGarage() pour qu'elle permette la saisie de 1 à 20 garages successivement. La variable nommée nbrGarage contiendra le nombre de garage créé.
- Transformez la fonction nommée afficheGarage() pour qu'elle affiche toutes les informations sur les garages et les voitures qui ont été saisies avec la fonctions saisieGarage(). La variable nbrGarage permet de savoir combien de garage ont été préalablement saisie.



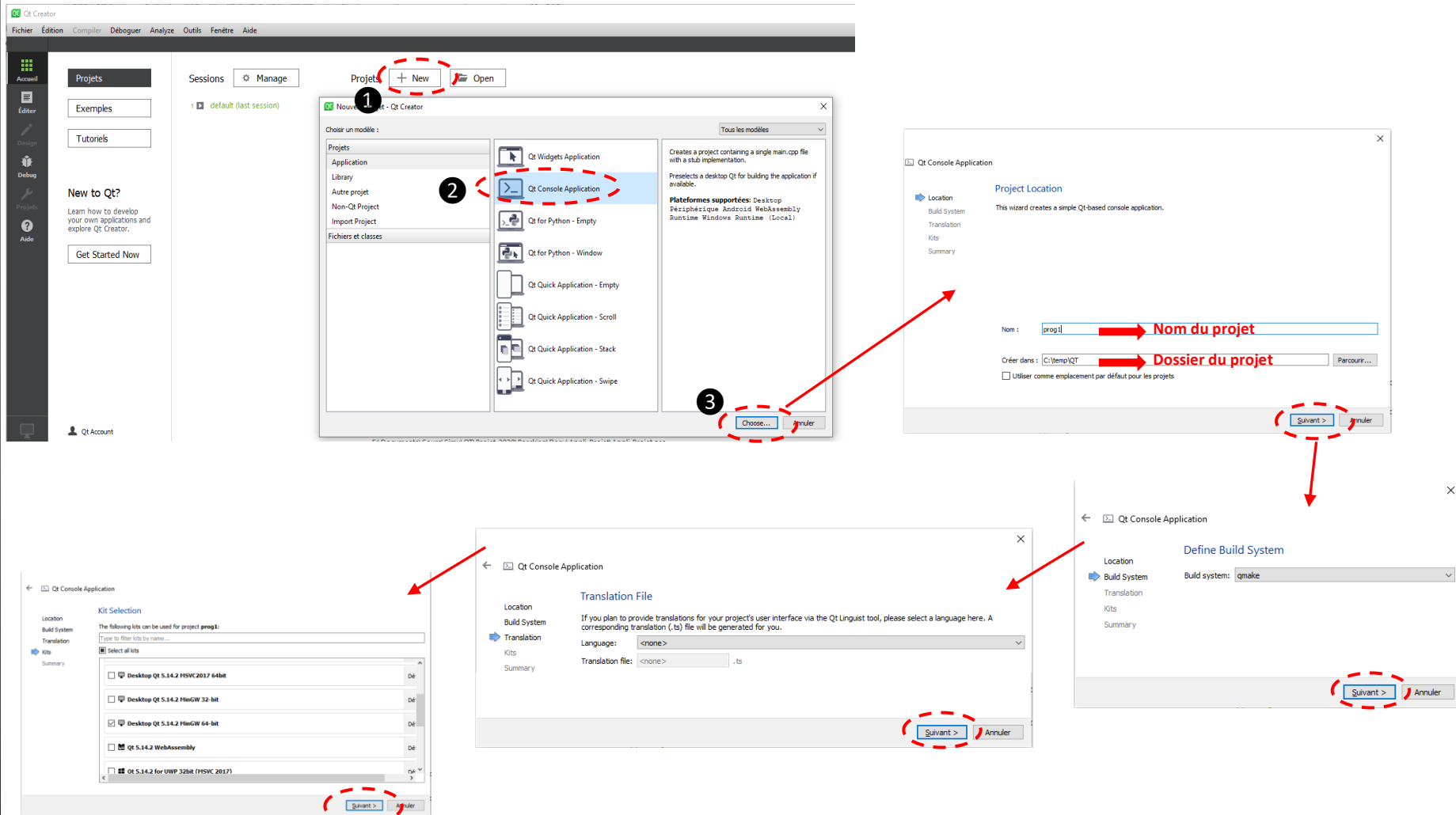
- Complétez le compte-rendu de mesures.

5. ANNEXE

Création d'un projet de type console :

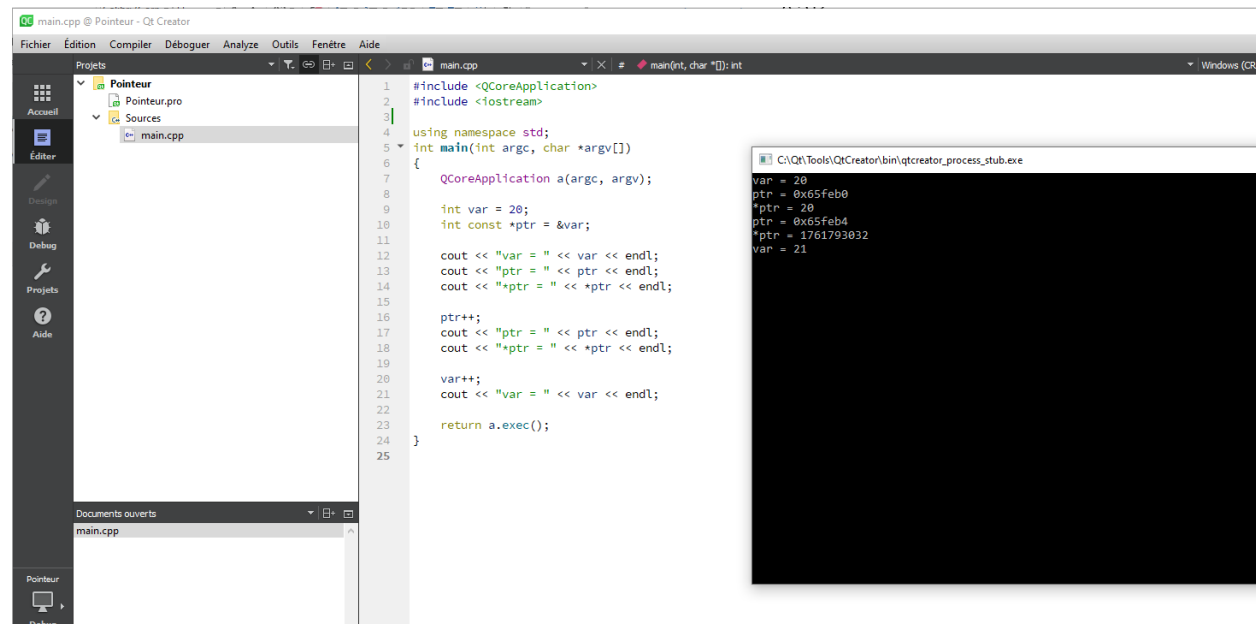
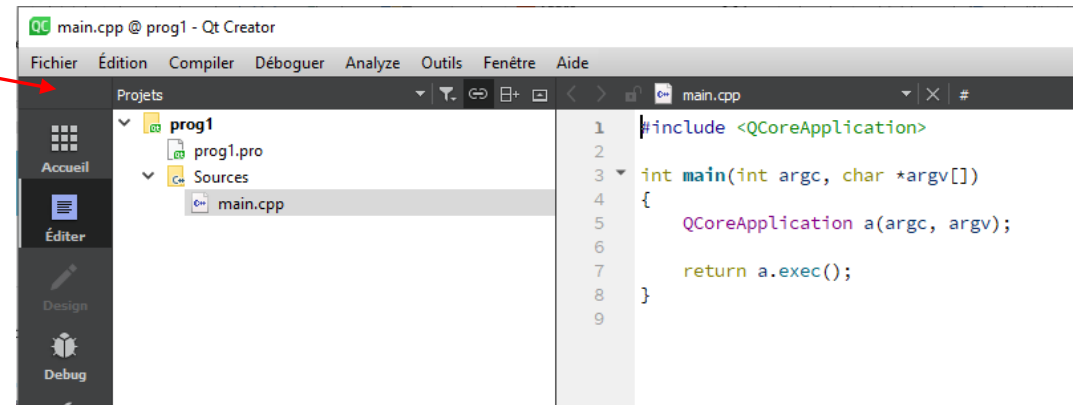
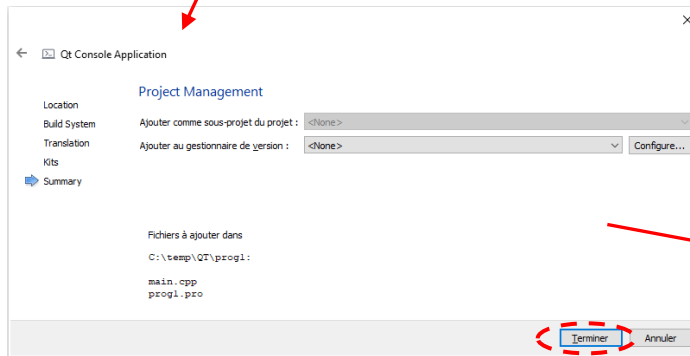
Introduction

1. DEFINITION D'UNE STRUCTURE
2. INSTANTIATION DE STRUCTURE
3. UTILISATION DES STRUCTURES
4. FINALISATION DU PROGRAMME
5. Annexes



Introduction

1. DEFINITION D'UNE STRUCTURE
2. INSTANTIATION DE STRUCTURE
3. UTILISATION DES STRUCTURES
4. FINALISATION DU PROGRAMME
5. Annexes



Ajout d'un fichier d'entête .h au projet :

main.cpp @ allocation_dynamique - Qt Creator

Fichier Édition Compiler Débugger Analyze Outils Fenêtre Aide

Nouveau fichier ou projet... Ctrl+N

Ouvrir un fichier ou projet... Ctrl+O

Ouvrir le fichier avec...

Fichiers récents

Projets récents

Sessions

Close Project "allocation_dynamique"

Close All Files in Project "allocation_dynamique"

New File or Project - Qt Creator

Choisir un modèle :

Projets

Application

Library

Autre projet

Non-Qt Project

Import Project

Fichiers et classes

C++

Modeling

Qt

GLSL

C++ Class

C++ Source File

C++ Header File

Creates a C++ header file and a project.

Plateformes supportées : Linux Device : WebAssembly Runtime (Local)

C++ Header File

Location

Summary

Nom : NomDuProjet → Nom du fichier

Chemin : C:\temp\qt → Dossier du projet

Le contenu de mon fichier d'entête.

#ifndef nomDeFichier_H

#define nomDeFichier_H

#endif // nomDeFichier_H

#include <QCoreApplication>

#include <iostream>

#include "nomDeMonFichier.h"

using namespace std;

int main(int argc, char *argv[])

{

Project Management

Location

Summary

Ajouter au projet : allocation_dynamique.pro → Nom du projet.pro

Ajouter au gestionnaire de version : <clone>

Fichiers à ajouter dans

E:\Documents\Bleu\QT\Console\Allocation_dynamique\allocation_dynamique :

test.h

Terminer Annuler

Introduction

1. DEFINITION D'UNE
STRUCTURE2. INSTANTIATION
DE STRUCTURE3. UTILISATION DES
STRUCTURES4. FINALISATION DU
PROGRAMME

5. Annexes