

(INSAM)

BTS Niveau 2

MSI236 - PROGRAMMATION I

30 THÈMES DE PROJETS EN LANGAGE C

Contrôle Continu : 20 points

Travaux Pratiques : 20 points

Enseignant : Ing. Steve Junior BOUSSA

Année académique 2024-2025

INTRODUCTION ET CONSIGNES

Ce document présente 30 thèmes de projets en langage C destinés aux étudiants de BTS Niveau 2. Chaque étudiant doit choisir UN thème parmi les 30 proposés. Tous les thèmes ont été conçus avec un niveau de difficulté équivalent.

Objectifs pédagogiques

- Maîtriser les concepts fondamentaux du langage C (variables, boucles, conditions)
- Manipuler les types composés (tableaux, structures, chaînes de caractères)
- Utiliser les pointeurs et l'allocation dynamique
- Créer et utiliser des fonctions
- Gérer les fichiers pour la persistance des données
- Organiser un projet en programmation modulaire

Consignes générales

1. Chaque étudiant choisit UN seul thème (pas de doublons dans la classe)
2. L'éditeur et l'environnement de développement sont libres (Code::Blocks, Dev-C++, VS Code, etc.)
3. Le code doit être propre, commenté et bien structuré
4. Un rapport technique doit accompagner le projet
5. Une soutenance orale sera organisée pour la présentation

Critères d'évaluation

Critère	CC (/20)	TP (/20)
Fonctionnalités implémentées	6 pts	8 pts
Qualité du code (commentaires, structure)	4 pts	4 pts
Interface utilisateur et ergonomie	3 pts	3 pts
Rapport et documentation	4 pts	2 pts
Soutenance et maîtrise du sujet	3 pts	3 pts

LES 30 THÈMES DE PROJETS

Projet 1 : Vérificateur de Palindrome

Description : Un palindrome est un mot, une phrase ou un nombre qui se lit de la même façon de gauche à droite et de droite à gauche.

Exemple : « RADAR », « KAYAK », « 12321 » sont des palindromes. « BONJOUR » n'en est pas un car lu à l'envers donne « RUOJNOB ».

Fonctionnalités attendues : Saisie d'un mot ou phrase, nettoyage (suppression espaces/accents), vérification, affichage du résultat.

Projet 2 : Jeu du Pendu

Description : Le pendu est un jeu de devinette où le joueur doit trouver un mot caché lettre par lettre. À chaque erreur, une partie du pendu est dessinée.

Exemple : Mot à deviner : INSAM → Affichage initial : _ _ _ _ _ → Le joueur propose 'A' → Affichage : _ _ _ A _ → Erreur avec 'Z' → Perd une vie.

Fonctionnalités attendues : Choix aléatoire d'un mot, affichage masqué, gestion des lettres proposées, compteur d'erreurs, dessin ASCII du pendu.

Projet 3 : Convertisseur de Bases Numériques

Description : Les ordinateurs utilisent différentes bases : binaire (base 2), octale (base 8), décimale (base 10), hexadécimale (base 16).

Exemple : Le nombre décimal 255 équivaut à 11111111 en binaire, 377 en octal, et FF en hexadécimal.

Fonctionnalités attendues : Saisie d'un nombre, choix de la base source et cible, conversion, affichage du résultat.

Projet 4 : Gestionnaire de Contacts

Description : Un carnet d'adresses numérique permettant de stocker et gérer les informations de vos contacts (nom, téléphone, email).

Exemple : Ajouter : Jean Dupont, 699123456, jean@email.com → Rechercher 'Dupont' → Affiche les informations de Jean.

Fonctionnalités attendues : Ajouter, modifier, supprimer, rechercher un contact, sauvegarder dans un fichier, charger depuis un fichier.

Projet 5 : Calculatrice Scientifique

Description : Une calculatrice capable d'effectuer des opérations mathématiques avancées : racine carrée, puissances, fonctions trigonométriques.

Exemple : $\sqrt{16} = 4$, $2^8 = 256$, $\sin(90^\circ) = 1$, $\cos(0^\circ) = 1$, $\log(100) = 2$.

Fonctionnalités attendues : Menu de sélection, opérations de base (+, -, *, /), puissance, racine, trigonométrie, logarithme, historique des calculs.

Projet 6 : Jeu du Morpion (Tic-Tac-Toe)

Description : Un jeu à deux joueurs sur une grille 3x3 où chacun place alternativement son symbole (X ou O). Le premier à aligner 3 symboles gagne.

Exemple : Joueur 1 (X) joue en case [1,1] → Joueur 2 (O) joue en case [0,0] → Continuer jusqu'à victoire ou match nul.

Fonctionnalités attendues : Affichage de la grille, saisie des coups, vérification de victoire (lignes, colonnes, diagonales), détection match nul.

Projet 7 : Générateur de Mot de Passe

Description : Un outil qui crée des mots de passe sécurisés en combinant lettres majuscules, minuscules, chiffres et caractères spéciaux.

Exemple : Longueur : 12, Options : majuscules + chiffres + spéciaux → Résultat : 'K9#mP2xL@4vB'

Fonctionnalités attendues : Choix de la longueur, sélection des types de caractères, génération aléatoire, évaluation de la force du mot de passe.

Projet 8 : Convertisseur de Devises

Description : Un programme qui convertit des montants d'une devise à une autre selon des taux de change prédéfinis.

Exemple : 100 EUR → USD avec taux 1.08 = 108 USD. 50000 FCFA → EUR avec taux 0.00152 = 76 EUR.

Fonctionnalités attendues : Menu des devises (EUR, USD, FCFA, GBP...), saisie du montant, conversion, affichage formaté.

Projet 9 : Système de Gestion de Notes

Description : Un programme pour gérer les notes des étudiants : saisie des notes, calcul des moyennes, classement.

Exemple : Étudiant : Marie, Notes : 15, 12, 18, 14 → Moyenne : 14.75 → Mention : Assez Bien.

Fonctionnalités attendues : Saisie des étudiants et notes, calcul moyenne par étudiant, moyenne de classe, classement, export fichier.

Projet 10 : Chiffrement de César

Description : Le chiffrement de César est une technique où chaque lettre est remplacée par une lettre décalée d'un nombre fixe dans l'alphabet.

Exemple : Décalage de 3 : 'BONJOUR' devient 'ERQMRXU'. A→D, B→E, C→F... Pour déchiffrer, on fait l'inverse.

Fonctionnalités attendues : Saisie du message, choix du décalage, chiffrement, déchiffrement, gestion des majuscules/minuscules.

Projet 11 : Jeu du Nombre Mystère

Description : L'ordinateur choisit un nombre aléatoire et le joueur doit le deviner. À chaque essai, l'ordinateur indique 'plus grand' ou 'plus petit'.

Exemple : Nombre mystère : 42. Essai 1 : 50 → 'Trop grand'. Essai 2 : 30 → 'Trop petit'. Essai 3 : 42 → 'Bravo !'

Fonctionnalités attendues : Génération aléatoire, saisie des propositions, indices, compteur d'essais, niveaux de difficulté, meilleurs scores.

Projet 12 : Tri de Tableaux

Description : Implémentation de différents algorithmes de tri pour ordonner des éléments (nombres ou textes) dans un ordre croissant ou décroissant.

Exemple : Tableau : [64, 34, 25, 12, 22, 11, 90] → Après tri croissant : [11, 12, 22, 25, 34, 64, 90]

Fonctionnalités attendues : Saisie du tableau, choix de l'algorithme (bulle, sélection, insertion), affichage étape par étape, comparaison des performances.

Projet 13 : Analyseur de Texte

Description : Un programme qui analyse un texte et fournit des statistiques : nombre de mots, caractères, phrases, fréquence des lettres.

Exemple : Texte : 'Bonjour INSAM' → 2 mots, 12 caractères (sans espaces), lettre 'N' apparaît 2 fois.

Fonctionnalités attendues : Saisie ou chargement de texte, comptage mots/caractères/lignes, fréquence des lettres, mot le plus long.

Projet 14 : Gestion de Bibliothèque

Description : Un système pour gérer les livres d'une bibliothèque : ajout, suppression, emprunt, retour, recherche.

Exemple : Ajouter : 'Le Petit Prince', Antoine de Saint-Exupéry, 1943 → Emprunter par 'Jean' → Statut : Emprunté.

Fonctionnalités attendues : Structure livre (titre, auteur, année, disponibilité), CRUD, recherche, liste des emprunts, sauvegarde fichier.

Projet 15 : Calcul de Factorielle et Fibonacci

Description : La factorielle de n (n!) est le produit de tous les entiers de 1 à n. Fibonacci est une suite où chaque terme est la somme des deux précédents.

Exemple : $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$. Fibonacci : 0, 1, 1, 2, 3, 5, 8, 13, 21... (0+1=1, 1+1=2, 1+2=3...)

Fonctionnalités attendues : Calcul factorielle (itératif et récursif), suite de Fibonacci, affichage des n premiers termes, vérification débordement.

Projet 16 : Convertisseur de Températures

Description : Conversion entre les différentes échelles de température : Celsius (°C), Fahrenheit (°F), Kelvin (K).

Exemple : $0^{\circ}\text{C} = 32^{\circ}\text{F} = 273.15\text{K}$. $100^{\circ}\text{C} = 212^{\circ}\text{F} = 373.15\text{K}$. Formule : $F = C \times 9/5 + 32$

Fonctionnalités attendues : Menu de conversion, saisie de la valeur, conversion dans les trois unités, tableau de correspondances.

Projet 17 : Gestion de Stock

Description : Un système de gestion d'inventaire pour suivre les produits : quantités, prix, alertes de stock bas.

Exemple : *Produit : Cahier, Quantité : 50, Prix : 500 FCFA → Vente de 10 → Nouveau stock : 40. Alerte si stock < 10.*

Fonctionnalités attendues : Ajout produit, mise à jour stock, vente, achat, alerte stock minimum, rapport d'inventaire, sauvegarde.

Projet 18 : Jeu Pierre-Papier-Ciseaux

Description : Un jeu classique où Pierre bat Ciseaux, Ciseaux bat Papier, et Papier bat Pierre. Le joueur affronte l'ordinateur.

Exemple : *Joueur : Pierre, Ordinateur : Ciseaux → Le joueur gagne ! Score : Joueur 1 - Ordinateur 0.*

Fonctionnalités attendues : Choix du joueur, choix aléatoire de l'ordinateur, détermination du vainqueur, score cumulé, partie en plusieurs manches.

Projet 19 : Calendrier et Jour de la Semaine

Description : Un programme qui affiche le calendrier d'un mois donné et détermine le jour de la semaine pour une date quelconque.

Exemple : *Date : 01/01/2025 → Mercredi. Afficher le calendrier de Janvier 2025 avec tous les jours alignés.*

Fonctionnalités attendues : Saisie date, calcul jour de semaine (algorithme de Zeller), affichage calendrier mensuel, gestion années bissextiles.

Projet 20 : Chronomètre et Minuterie

Description : Un programme qui simule un chronomètre (compte le temps écoulé) et une minuterie (compte à rebours).

Exemple : *Chronomètre : Start → 00:00:00 → 00:01:30 → Stop. Minuterie : 05:00 → 04:59 → ... → 00:00 → Alarme !*

Fonctionnalités attendues : Mode chronomètre (start/stop/reset), mode minuterie (définir durée), affichage temps, signal fin de minuterie.

Projet 21 : Simulateur de Machine à Sous

Description : Un jeu de casino simplifié où trois rouleaux affichent des symboles aléatoires. Trois symboles identiques = jackpot !

Exemple : Mise : 100 FCFA → Résultat : [][] → Jackpot ! Gain : 1000 FCFA. [][*] → Perdu.

Fonctionnalités attendues : Gestion du solde, mise, tirage aléatoire des symboles, calcul des gains selon combinaisons, affichage ASCII.

Projet 22 : Gestionnaire de Tâches (To-Do List)

Description : Une application pour gérer ses tâches quotidiennes : ajouter, marquer comme terminée, supprimer, afficher.

Exemple : Ajouter : 'Réviser le C' → Ajouter : 'Faire le TP' → Marquer 'Réviser le C' comme terminé → ✓ Réviser le C

Fonctionnalités attendues : Ajout tâche avec priorité, modification statut, suppression, tri par priorité/date, sauvegarde fichier.

Projet 23 : Vérificateur de Nombre Premier

Description : Un nombre premier est un nombre supérieur à 1 qui n'est divisible que par 1 et lui-même.

Exemple : 2, 3, 5, 7, 11, 13 sont premiers. 4 n'est pas premier (divisible par 2). 9 n'est pas premier (divisible par 3).

Fonctionnalités attendues : Vérifier si un nombre est premier, afficher les n premiers nombres premiers, trouver les premiers dans un intervalle.

Projet 24 : Système de Vote Électronique

Description : Un programme simulant un système de vote : inscription des candidats, vote des électeurs, comptage et résultats.

Exemple : Candidats : Alice, Bob, Charlie → Vote 1 : Alice, Vote 2 : Bob, Vote 3 : Alice → Résultat : Alice 2 voix (66.7%)

Fonctionnalités attendues : Inscription candidats, vote unique par électeur, comptage des voix, pourcentages, détection égalité, affichage graphique.

Projet 25 : Convertisseur Binaire-Texte

Description : Chaque caractère a un code ASCII qui peut être représenté en binaire. Ce programme convertit du texte en binaire et inversement.

Exemple : 'A' = 65 en ASCII = 01000001 en binaire. 'Hi' = '01001000 01101001'

Fonctionnalités attendues : Texte vers binaire, binaire vers texte, affichage table ASCII, support des caractères spéciaux.

Projet 26 : Calculateur d'IMC

Description : L'Indice de Masse Corporelle (IMC) évalue la corpulence. Formule : $IMC = \text{poids(kg)} / \text{taille(m)}^2$

Exemple : Poids : 70kg, Taille : 1.75m → $IMC = 70 / 1.75^2 = 22.9$ → 'Poids normal' (entre 18.5 et 25).

Fonctionnalités attendues : Saisie poids/taille, calcul IMC, interprétation (maigreur, normal, surpoids, obésité), conseils, historique.

Projet 27 : Générateur de Tableaux de Multiplication

Description : Un programme qui génère et affiche les tables de multiplication de 1 à n, formatées proprement.

Exemple : Table de 7 : $7 \times 1 = 7$, $7 \times 2 = 14$, $7 \times 3 = 21 \dots 7 \times 10 = 70$. Afficher plusieurs tables côte à côte.

Fonctionnalités attendues : Choix de la table, affichage formaté, quiz interactif, affichage grille complète 1-10.

Projet 28 : Système de Réservation

Description : Un système de réservation simplifié (restaurant, cinéma, hôtel) avec gestion des créneaux disponibles.

Exemple : Créneaux restaurant : 12h, 13h, 14h, 19h, 20h. Réserver 19h au nom de 'Dupont' → Créneau 19h : OCCUPÉ

Fonctionnalités attendues : Affichage créneaux disponibles, réservation avec nom/date, annulation, liste des réservations, sauvegarde.

Projet 29 : Jeu du Puissance 4

Description : Un jeu à deux joueurs sur une grille 7x6. Les jetons tombent en bas de la colonne. Le premier à aligner 4 jetons gagne.

Exemple : Joueur 1 place en colonne 3 → Le jeton tombe en bas. Aligner 4 horizontalement, verticalement ou en diagonale pour gagner.

Fonctionnalités attendues : Affichage grille, gestion de la gravité des jetons, détection victoire (4 directions), détection grille pleine.

Projet 30 : Quiz de Culture Générale

Description : Un jeu de questions-réponses avec score. Les questions sont stockées dans un fichier ou un tableau.

Exemple : Question : 'Capitale du Cameroun ?' → A) Douala B) Yaoundé C) Bafoussam → Réponse : B → Correct ! +10 points

Fonctionnalités attendues : Banque de questions (fichier), questions aléatoires, QCM ou réponse libre, score, classement, ajout de questions.

BONUS ET EXTENSIONS

Pour obtenir des points bonus (jusqu'à +2 points), vous pouvez implémenter les extensions suivantes :

- Sauvegarde persistante des données dans un fichier
- Interface en couleurs (utilisation de bibliothèques comme ncurses ou conio.h)
- Mode multijoueur local
- Gestion avancée des erreurs de saisie
- Organisation du code en plusieurs fichiers (programmation modulaire)
- Utilisation de Makefile pour la compilation

STRUCTURE DU RAPPORT ATTENDU

6. Page de garde (nom, filière, thème choisi)
7. Introduction et présentation du projet
8. Analyse et conception (diagrammes, algorithmes)
9. Réalisation (extraits de code commentés)
10. Tests et résultats (captures d'écran)
11. Difficultés rencontrées et solutions
12. Conclusion et perspectives d'amélioration
13. Annexes (code source complet)

« La programmation, c'est comme la cuisine : on suit une recette, mais le résultat dépend du chef ! »

Bon courage à tous !

Ing. Steve Junior BOUSSA