Trabalho Arq. de Computadores II

Wolfenstein

GRUPO:

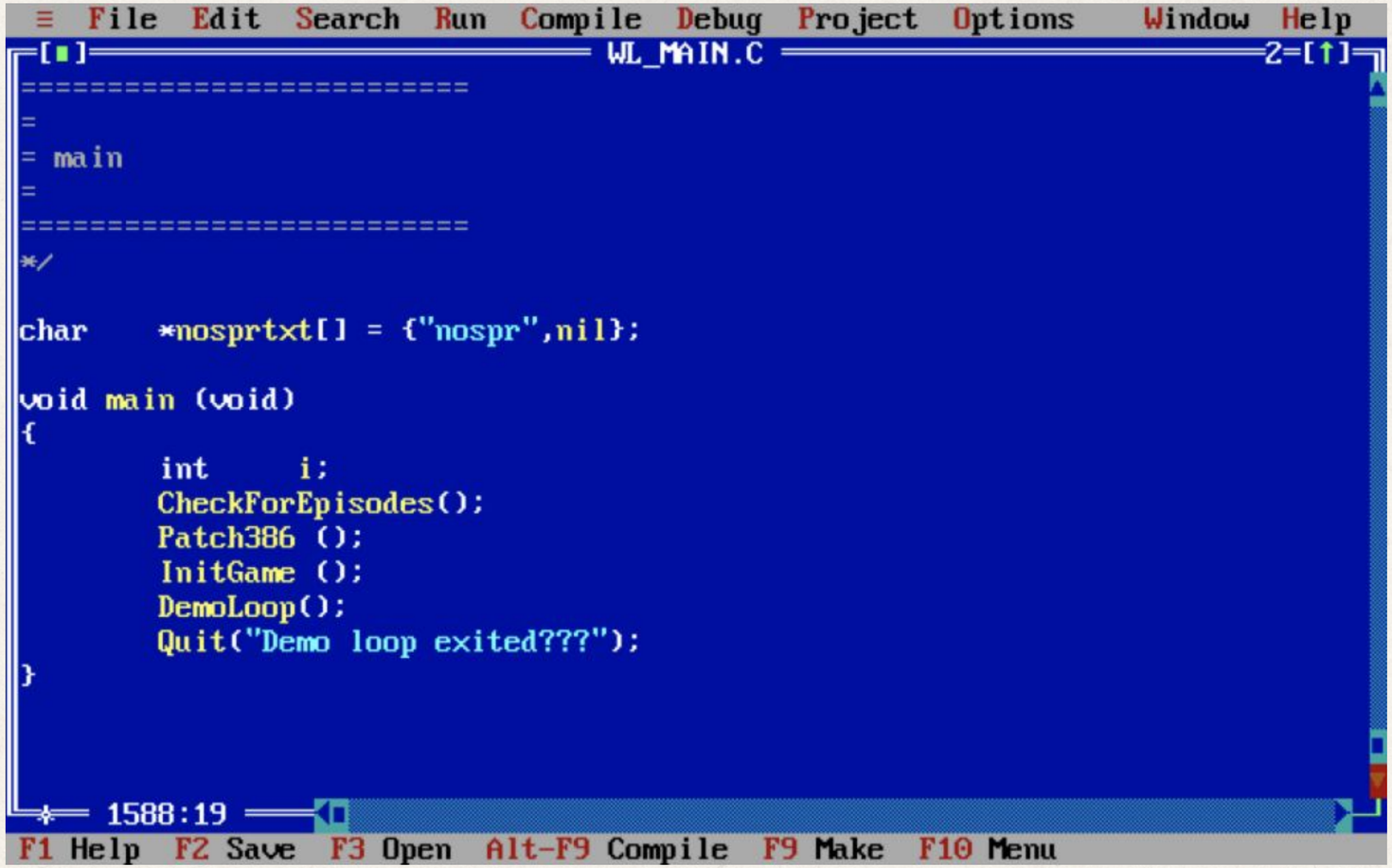
- Cauã Cola Zandonadi
- Davi Ribeiro Carvalho
- Guilherme Sousa Fagundes
- Giseli Rosa Lourenço
- Matheus Miranda Griffo

Programação e Ambiente de desenvolvimento

- O DESENVOLVIMENTO FOI FEITO COM BORLAND C++ 3.1, MAS A LINGUAGEM USADA FOI C.

- O AMBIENTE DA BORLAND INTEGRAVA FERRAMENTAS DE COMPILAÇÃO, EXECUÇÃO E DEPURAÇÃO, DENTRO DO PRÓPRIO AMBIENTE INTEGRADO.

- POR PADRÃO RODAVA NO MODO VGA 3, OFERECENDO UMA TELA COM 80 CARACTERES DE LARGURA E 25 DE ALTURA.



Divisão das Responsabilidades

John Carmack	Cuidou do código de tempo de execução
John Romero	Programou muitas das ferramentas (editor de mapas TED5, compactador de recursos IGRAB, compactador de som MUSE)
Jason Blochowiak	Escreveu subsistemas importantes do jogo (gerenciador de entrada, gerenciador de páginas, gerenciador de som, gerenciador de usuários).

Solução Técnica

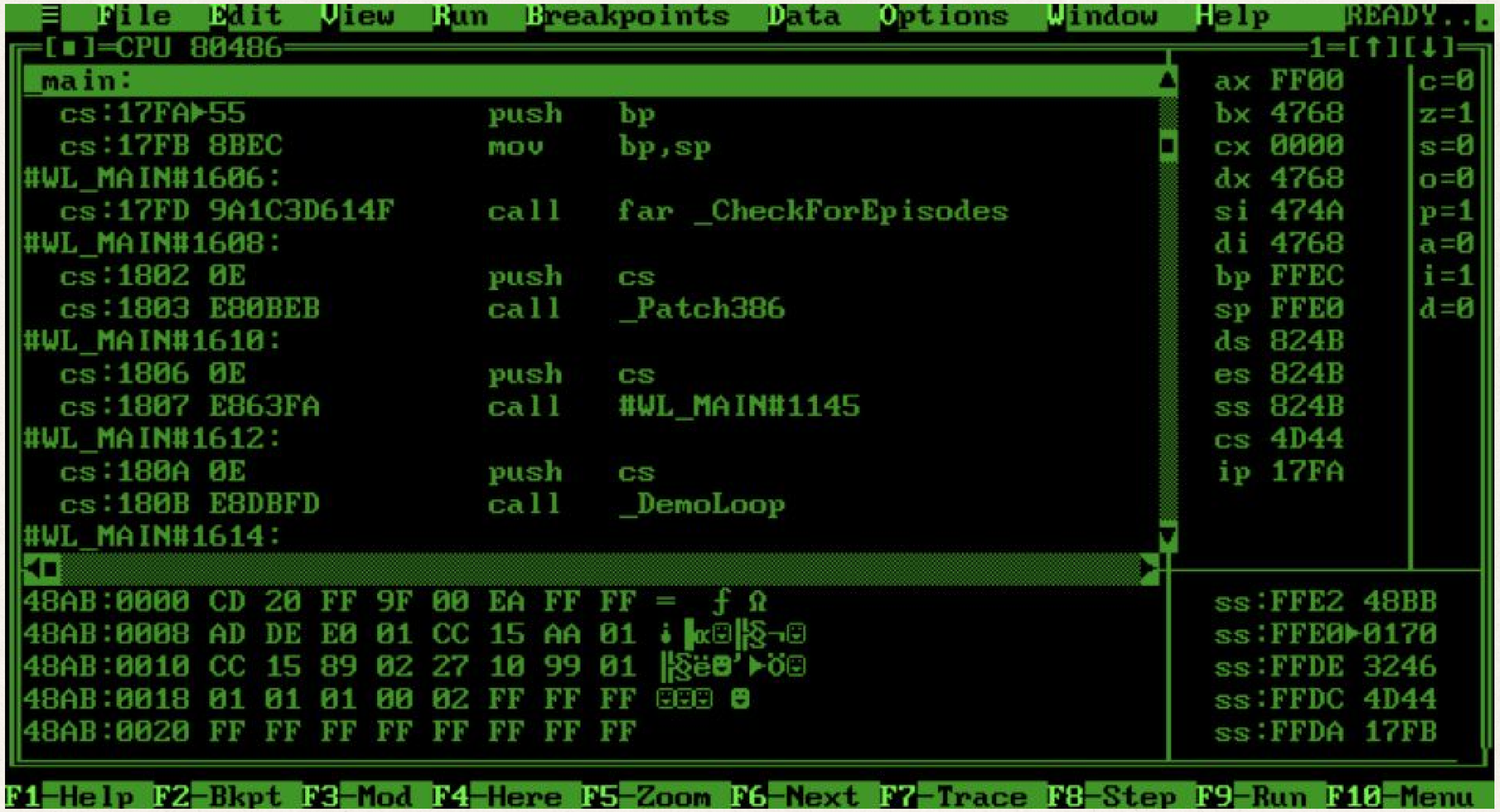
PARA CONTORNAR AS LIMITAÇÕES DE VISUALIZAÇÃO DAS TELAS CTR DURANTE O PROCESSO DE DEPURAÇÃO CONSISTIU NA UTILIZAÇÃO DE DOIS MONITORES SIMULTANEAMENTE.

- A **PLACA VGA** OPERAVA EM MODO GRÁFICO (MODO 13H), UTILIZANDO O ENDEREÇO 0XA0000 PARA O FRAMEBUFFER, ISTO É, PARA EXECUTAR O JOGO NORMALMENTE.

- A **PLACA MDA** OPERAVA EM MODO TEXTO MONOCROMÁTICO, ACESSANDO O ENDEREÇO 0XB8000 PARA COLETA DE DADOS.

ESSA SEPARAÇÃO POSSIBILITAVA A **EXECUÇÃO DO JOGO** EM MODO GRÁFICO SEM INTERFERÊNCIAS, **AO MESMO TEMPO** EM QUE FERRAMENTAS COMO O TURBO DEBUGGER 386 EXIBIAM INFORMAÇÕES DE **DEPURAÇÃO** NO MONITOR SECUNDÁRIO.

Solução Técnica



Tela MDA com o depurador Borland 3.1

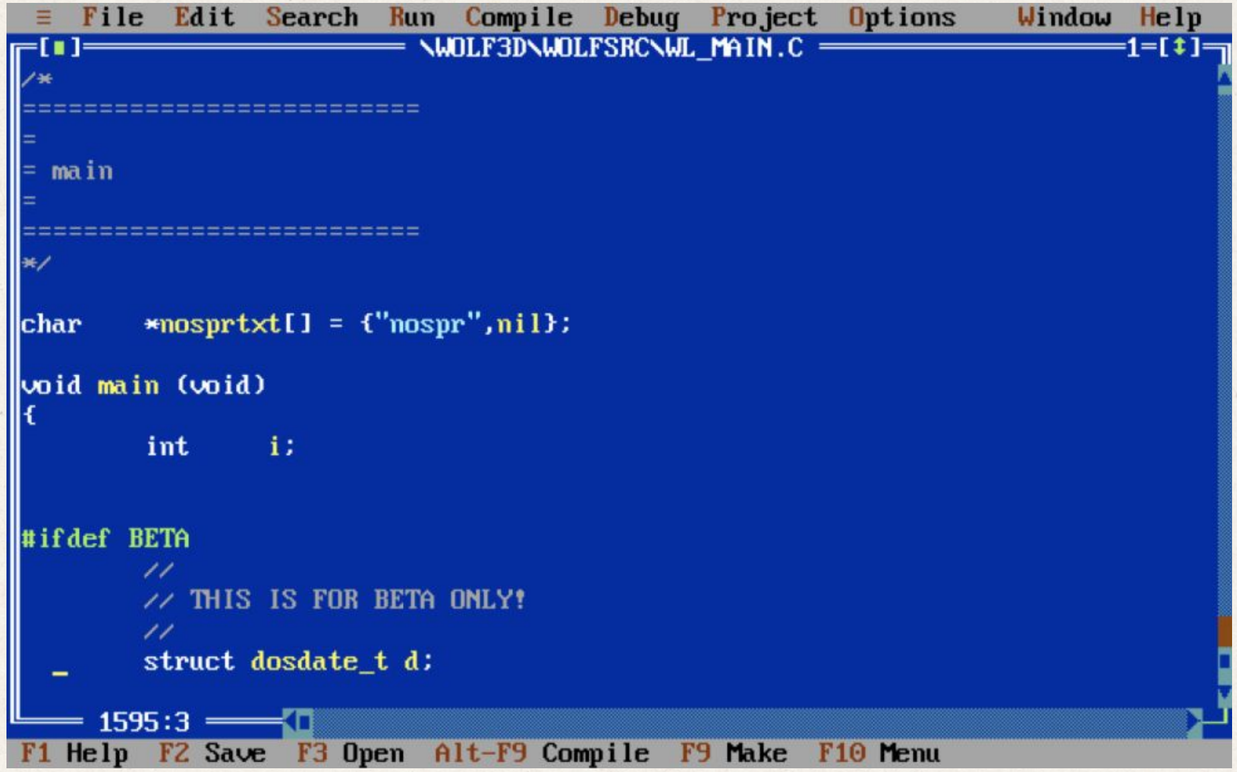
Solução Técnica

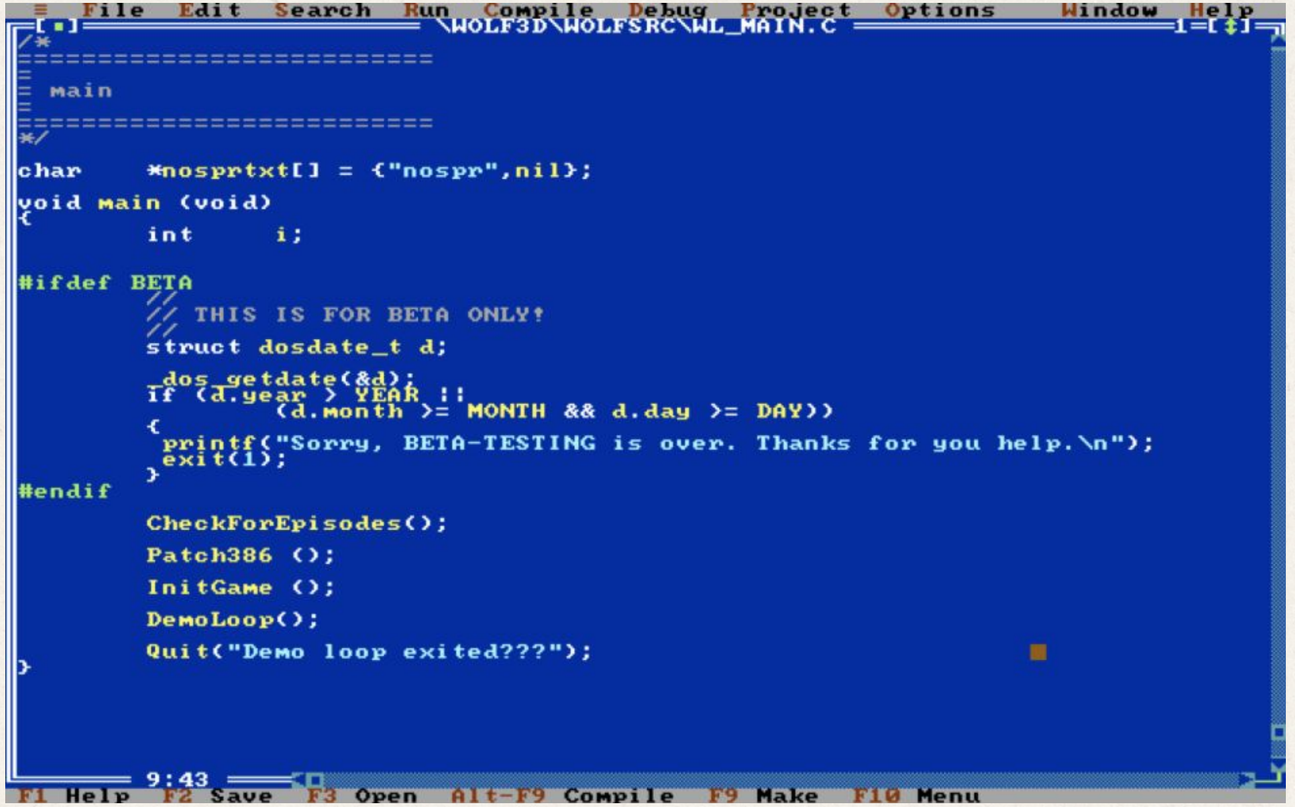
Tela VGA executando jogo

Solução Técnica

Outra maneira de melhorar o espaço na tela era usar o modo de texto de "alta resolução" 50x80.

Exemplo: O arquivo WL_MAIN.C aberto em ambos os modos demonstra a compensação entre legibilidade e visibilidade.





Assets Gráficos

- ADRIAN CARMACK PRODUZIU TODOS OS ASSETS GRÁFICOS.

- TODO O TRABALHO FOI FEITO NO DELUXE PAINT, EDITOR GRÁFICO EM BITMAP DESENVOLVIDO PELA ELETROTNIC ARTS.

- NÃO UTILIZARAM FERRAMENTAS DE SCANNEAMENTO PRESENTES NA EPOCA, TODOS OS ASSETS FORAM FEITOS MANUALMENTE, PIXEL POR PIXEL.

- ADRIAN DESENHAVA NA RESOLUÇÃO NATIVA DO JOGO, 320×200 PIXELS, DIRETAMENTE COM MOUSE.

Assets Gráficos

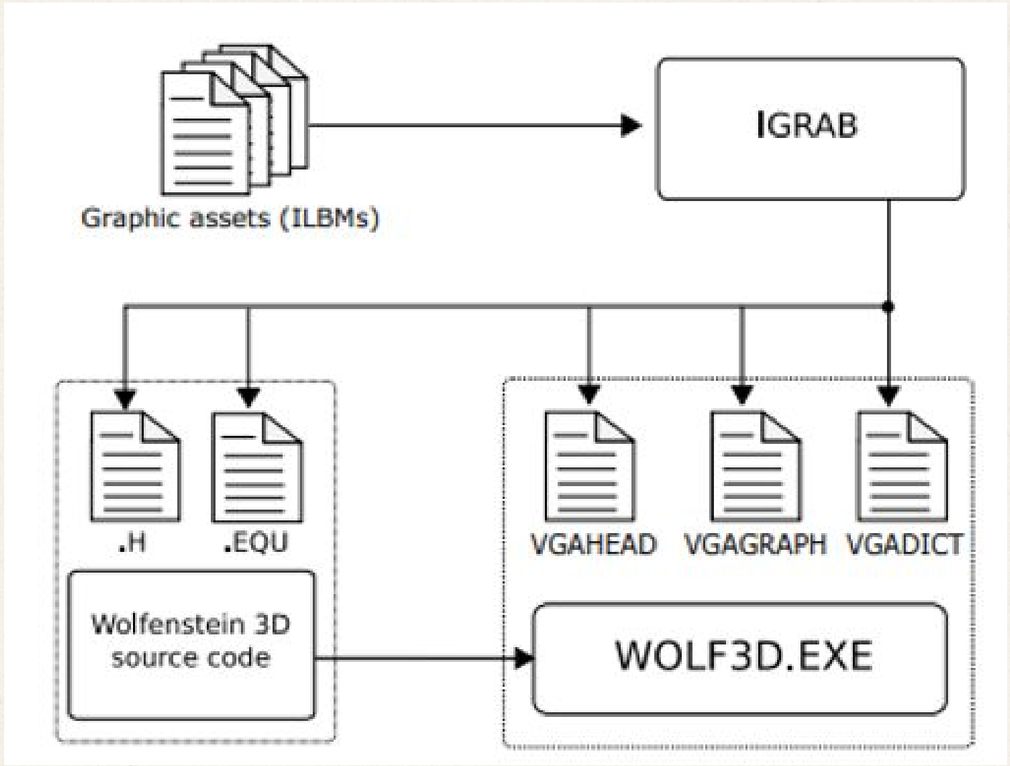
- O VGA NO MODO 13H NÃO OPERAVA NO PADRÃO 24BITS-RGB, EM VEZ DISSO, ERA UTILIZADO UMA PALETA COM 256 CORES, ONDE CADA PIXEL POSSUIA UM INDICE QUE APONTAVA PARA A PALETA.

- ADRIAN CRIOU A PALETA COM AS 256 CORES E AS UTILIZOU EM TODOS OS ASSETS DO JOGO

- CADA PIXEL CONTIA UM INDICE QUE INDICAVA PARA A PALETA.

Worfkflow de Assets

- DEPOIS DE GERADOS, OS ASSETS GRÁFICOS, JÁ NO FORMATO ILBM (PADRÃO DO DELUXE PAINT), ERAM EMPACOTADOS EM UM ARQUIVO.
- ERA ENTÃO GERADO UM ARQUIVO HEADER EM C, CONTENDO OS IDS DE CADA ASSET.
- A ENGINE ENTÃO, REFERÊNCIAVA CADA ASSET USANDO DIRETAMENTE ESSES IDS.
- ESTE PROCESSO ERA FEITO UTILIZANDO A FERRAMENTA IGRAB.



Worfkflow de Assets

- NO CÓDIGO DA ENGINE, A UTILIZAÇÃO DOS ASSETS É FEITA USANDO UM ENUM.
- ESSE ENUM, É UM OFFSET DO ARQUIVO HEADER, QUE POR SUA VEZ, É UM OFFSET DO ARQUIVO DE ASSETS.
- COM ESSAS CAMADAS INDIRETAS, ERA POSSÍVEL REGENERAR E REORDENAR ASSETS A VONTADE, SEM MEXER NO CÓDIGO FONTE.

```
//////////////////////////////////////
//
// Graphics .H file for .WL1
// IGRAB-ed on Sun May 03 01:19:32 1992
//
//////////////////////////////////////

typedef enum {
    // Lump Start
    H_BJPIC=3,
    H_CASTLEPIC,                    // 4
    H_KEYBOARDPIC,                  // 5
    H_JOYPIC,                       // 6
    H_HEALPIC,                      // 7
    H_TREASUREPIC,                  // 8
    H_GUNPIC,                       // 9
    H_KEYPIC,                       // 10
    H_BLAZEPIC,                     // 11
    H_WEAPON1234PIC,                // 12
    H_WOLFLOGOPIC,                  // 13
    ...
    PAUSEDPIC,                      // 140
    GETPSYCHEDPIC,                  // 141
```

Encerramento