Design Description

Game summary: The player must collect wooden blocks to reach cookies on the kitchen counter before the babysitter wakes up. Some rooms or items can only be obtained by obtaining a different item first.

Every movement to another room is 1 time unit. Every act of inspection is 1 time unit.
Player has 30 time units before the babysitter wakes up.

One possible win strategy:

- Player's room:
    - Crib
        - Take squid and unicorn toys
    - Closet
        - Take wagon and wooden block
    - Drawer
        - Wear fuzzy socks
- Hallway: leave wooden block
- Living room: trade cat squid for keys
- Parents' room:
    - Closet: wooden block
- Parents' Bathroom:
    - Cabinet under sink: baby oil
- Sister's room:
    - Crib: take wheel and wooden block
- Hallway: pick up wooden block
- Kitchen: get cookies

General menu:

- Inspect room
- Move to another room
- Manage objects

Space (abstract class)

Member variables:

- Space *top
- Space *right
- Space *right2
- Space *bottom
- Space *bottom2
- Space *left
- string name: name of space
- int numNeighbors: number of available adjacent spaces
- list<int> items: list of items
- bool state: state of space that needs to be resolved before accessing
- ITEM solution: solution that resolves space state
- string passMsg: message that prints if state is resolved
- string failMsg: message that prints if state is not resolved

Member functions:

- Space()
  - Set all pointers to NULL and state to true
- Space(string)
  - Set all pointers to NULL and state to true
  - Sets name
- Space(string, int)
  - Set all pointers to NULL and state to true
  - Sets number of available adjacent neighbors
- Space(string, ITEM)
  - Set all pointers to NULL and state to false
  - Sets solution
- virtual ~Space() = 0: makes class abstract
- string getName: returns name of space
- int getNumNeighbors: returns number of available adjacent spaces
- void listConnectedSpaces: lists names of available adjacent spaces
- void addItem(ITEM): adds item to space's items list
- void removeItem(ITEM): removes item from space's items list
- int getItem(int): returns item from space's items list
- void listItems: list items
- int getNumItems: returns number of items available
- vector<Space*> POI: points of interest (possible inspection points) stored in a vector of Space pointers
- void setState(bool)
- bool getState
- void setStateDescription(string)
- void printStateDescription
- void setSolution(ITEM)
- int getSolution

- void setPassMsg(string) - void printPassMsg - void setFailMsg(string) - void printFailMsg

Bedroom (inherits from Space)
Member variables:
Member functions: - Bedroom() - Bedroom(string, num)

Bathroom (inherits from Space)
Member variables:
Member functions: - Bathroom() - Bathroom(string, num)

Hallway (inherits from Space)
Member variables:
Member functions: - Hallway() - Hallway(string, num)

LivingRoom (inherits from Space)
Member variables:
Member functions: - LivingRoom() - LivingRoom(string, num)

Kitchen (inherits from Space)
Member variables:
Member functions: - Kitchen() - Kitchen(string, num)

PointOfInterest (inherits from Space)
Member variables:
Member functions: - PointOfInterest() - PointOfInterest(string, ITEM)

Utility
enum ITEM { BLANKET, SQUID, UNICORN, WAGON3, BLOCK_A, BLOCK_B, BLOCK_C, HAT, SOCKS, MITTENS, KEY, OIL, WHEEL, BIB, DIAPER, BOTTLE, WAGON4, COOKIES, SQUEAKY, WAGON5, BLOCK3 } string displayItemName(int): displays item’s descriptive name int getInt(int, int): prompts user for an int with input validation char getYesOrNo(): prompts user for Y or N with input validation

Game
Member variables: - Space *playersRoom - Space *hallway - Space *sistersRom - Space *hallwayBath - Space *livingRoom - Space *parentsRoom - Space *parentsBath - Space *kitchen - Player player - Space *pLocation: player’s location

- int timer

Member functions:
- Game()
  - Displays introduction text
  - Creates new rooms in the house for Space pointers
  - Connects the rooms in the house
  - Adds items and sets up Space states
  - Sets pLocation to playersRoom
  - Sets timer to 25
- ~Game()
  - Sets Space pointers in each room's Space to NULL
  - Deletes each room
  - Sets each room to NULL
- play()
  - Plays the actual game
  - Calls displayInitialOptions while timer > 0 && player.getHasCookies() == false
  - Displays win or lose message after game ends
- displayInitialOptions
  - Prints map of house
  - Options
    - 1. Inspect the area
    - 2. Go to another area
    - 3. Manage my inventory
- inspect()
  - Calls inspect0 if items were dropped in the room
  - Lists possible Spaces to inspect and calls inspect2 to process user input
- inspect0
  - Handles case where items are dropped in the room
  - Prompts user to select item to pick up
- inspect2
  - Checks if Space being inspected has a state that needs to be resolved first
  - If no state needs to be resolved, call inspect3
- inspect3
  - Prompts user to select an item to pick up
- movePlayer: moves the player to user-selected location
- manageInventory: allows user to drop an item currently being held by user
- haveSpace(int, int): checks if user has space in inventory to take an item
- clear(): prints 100 newlines to "clear" the terminal screen
- map(): prints a map of the house and marks user's current location

Player

Member variables:
- int itemCount

- int cap: max number of items that can be held
    - Wooden blocks take up 3 capacity spot
    - All other items take up 1 capacity spot
    - default capacity: 2
    - 3-wheeled wagon capacity: 5
    - 4-wheeled wagon capacity: 10
- int numBlocks
- bool hasWagon
- bool hasWheel
- bool hasOil
- bool hasSqueaky
- bool hasCookies
- bool cap10Msg
- bool oilMsg
- bool wag5Msg

Member functions:

- Player()
    - Set itemCount to 0
    - Set cap to 2
    - Set numBlocks to 0
    - Set all flags to false
- int getItemCount
- void setCap
- int getCap
- list<int> items
- int getItem(int)
- void listItems
- void updateItems(int, bool)
- void setHasCookies(bool)
- bool getHasCookies

Hallway (Hallway object)

- Entrance to player's room, sister's room, living room, hallway bathroom, and parents' room

Parents' room (Bedroom object)

- State: Initially locked
    - Key is with cat in living room
- POIs (points of interest)
    - Bed: nothing
    - Closet: wooden block
- Entrance to parents' bathroom and hallway

Parents' bathroom (Bathroom object)

- Sink: nothing
- Shower: nothing
- Cabinet under sink:
    - Baby oil
- Entrance to parents' room

Hallway bathroom (Bathroom object)
- Dead end; black hole for time units
- Sink: nothing
- Shower: nothing
- Cabinet under sink:
    - Empty bottle of baby oil
- Entrance to hallway

Player's room (Bedroom object)
- Crib: blanket, squid made of yarn, fluffy unicorn toy
- Closet: 3-wheeled wagon, wooden block
- Drawer: hat, fuzzy socks, mittens
- Entrance to hallway

Little sister's bedroom (Bedroom object)
- State: Need to give sister unicorn for access
    - Unicorn is in player's crib
- Crib:
    - Squeaky wagon wheel inside crib
    - Wooden block under crib
- Drawer
    - Bib
    - Diapers
- Entrance to hallway

Living room (LivingRoom object)
- State: The only room with squeaky wooden floors
    - Need fuzzy socks in player's drawer
- Cat:
    - State: need to trade squid made of yarn in player's crib for parents' room key
- Babysitter: nothing
- Entrance to kitchen and hallway

Kitchen (Kitchen object)
- Counter:
    - State: Need 3 wooden blocks to reach cookies
- Entrance to living room

Testing

Test Case	Expected Outcome	Observed Outcome	Solution
Input validation for Start menu	1: proceed 2: exit program Other: Please enter an integer from 1 to 2:	1: proceed 2: exit program Other: Please enter an integer from 1 to 2:	
Input validation for Start menu	1: Lists possible inspect points/points of interest, if any 2: Lists possible adjacent rooms 3: Lists user's currently held items, if any Other: Please enter an integer from 1 to 3:	1: Lists possible inspect points/points of interest, if any 2: Lists possible adjacent rooms 3: Lists user's currently held items, if any Other: Please enter an integer from 1 to 3:	
Cancel option	Choosing cancel when inspecting or moving does not count towards timer	Timer decreases	Place timer-- in appropriate locations so timer is only affected when player inspects a point of interest or successfully moves
Movement stopped by failing Space state	If a player moves into a Space that requires resolving a state (i.e. need key for parents' room), movement is cancelled, and timer is not affected	Timer decreases by one	Place timer-- in appropriate locations so timer is only affected when player inspects a point of interest or successfully moves
Reaching default capacity of 2	You don't have enough space for that!	You don't have enough space for that!	
Taking block (size 3) in the beginning	You don't have enough space for that!	No error message Inventory count drops to -1	Check if player's current capacity minus current item count is 3 or more

Player obtains wagon	Inventory capacity increased to 5!	Inventory capacity increased to 5!	
Player obtains wheel while holding wagon	Inventory capacity increased to 10!	Inventory capacity increased to 10!	
Player obtains wheel, then obtains wagon	Inventory capacity increased to 5! You put the wheel on the wagon. Inventory capacity increased to 10!	Inventory capacity increased to 5! You put the wheel on the wagon. Inventory capacity increased to 10!	
Input validation for Anything else? (Y/N)	Y or y: repeat action N or n: proceed Other: Please select Y for Yes or N for No:	Y or y: repeat action N or n: proceed Other: Please select Y for Yes or N for No:	
Squeaky wagon in hallway	Your squeaky wagon woke up the babysitter! Game should end	Your squeaky wagon woke up the babysitter! Game continues	Change timer to 0 after this fail message prints
Enter living room without fuzzy socks	Your bare feet on the squeaky wooden floor woke the babysitter! Game should end	Your bare feet on the squeaky wooden floor woke the babysitter! Game continues	Change timer to 0 after this fail message prints
Take fuzzy socks	You put on Fuzzy socks. This does not count towards your inventory. Inventory count should not change	You put on Fuzzy socks. This does not count towards your inventory. Inventory count changes	Insert a conditional that does not change inventory when object is wearable
Drop wooden block in hallway	Space available in inventory should increase by 3	Space available in inventory increases by 3	
Drop wooden block in hallway, then manage inventory again	Block should not be in inventory	Block is not in inventory	
Drop wooden	Should get option to pick	Option to pick up block is	

block in hallway, and inspect hallway	up the block	given	
Enter living room with fuzzy socks	Your fuzzy socks make you a silent ninja baby! The wooden floor does not creak.	Your fuzzy socks make you a silent ninja baby! The wooden floor does not creak.	
Approach cat with squid toy	You trade Sammy the squid for a key.	You trade Sammy the squid for a key.	
Enter parents' room with key	You open the door with the key.	You open the door with the key.	
Go to sister's room with unicorn toy	You give your little sister Fluffy the unicorn. She lets you in her room.	You give your little sister Fluffy the unicorn. She lets you in her room.	
Obtain 3 blocks	Inventory should say Stairway of wooden blocks	Inventory says stairway of wooden blocks	
Get cookies	Your stairway of wooden blocks gets you to the cookies!	Your stairway of wooden blocks gets you to the cookies!	
Valgrind	No memory leaks	No memory leaks	